Author:
Stott, Harry B A

Title:
Modelling inertial particles in fluid flows. Efficient numerical approaches.

Modelling inertial particles in fluid flows.

Efficient numerical approaches.

By

HARRY STOTT

Department of Mechanical Engineering
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Engineering.

AUGUST 2018

Word count: 68500

Abstract

The dynamics of inertial particles have a profound influence in countless contexts in the natural and manmade world. Whether it be droplets of water advected in the atmosphere that cluster and collide to form rain, or volcanic ash clouds carried on the winds that ground aircraft and block out the sun, inertial particles are prolific in their influence on our lives.

An inertial particle is an object with material density different to the fluid in which it resides. This density difference causes heavy aerosols to be ejected from vortices, and light bubbles to migrate towards vortex centres. These behaviours characterise inertial particles and underpin complex particle processes such as clustering and droplet growth. Due to the breadth of important length scales in many two phase systems, different models are required for different parts of the scale space.

We present a hierarchy of modelling approaches that efficiently capture the inertial properties of particles and can be applied to a wide range of problems. Beginning with comparisons between Lagrangian particle trajectories and experimental results from the literature, we identify, and present solutions to, complications in numerical integration of the Maxey-Riley equation. The numerical model is applied in a paradigmatic vortical flow to help understand the influence of the Basset history force on particle trajectories. For applications with smaller particles, we develop an Eulerian description of the dynamics of particle ensembles. The method is a discretisation of a transfer operator that efficiently captures the important properties of large numbers of particles; its algorithmic complexity is independent of the number of particles, even when collisions are considered between them. Two methods for detecting particle collisions that dramatically improve on algorithmic complexity of pairwise detection schemes are presented and compared. Despite the philosophical difference between the Lagrangian and Eulerian approaches, comparisons between them produce remarkably similar results.

Our methods are then applied in a decaying turbulent flow to investigate sedimentation rates and particle clustering. Throughout, great care is taken to understand the nuances of particle dynamics, allowing parallels to be drawn between behaviours in steady and turbulent flows. From the perspective of the particles, we show that there is very little difference between particle clustering in steady and unsteady flows, provided a range of length scales are present in both.

Work is also conducted that explores the influence of the Basset history force in a range of contexts: vortex ejection, collision rates, and clustering. In decaying turbulence, the Basset force is shown to have a significant effect even for dense aerosols. With reference to these results, we discuss the decay rate of the Basset kernel relative to the decay rate of the turbulence, and question whether its form is appropriate in such contexts.

Dedication and acknowledgements

I would like to thank my supervisors, Dr. Andrew G. W. Lawrie and Dr. Robert Szalai, for their patience, knowledge and, towards the end, much needed words of encouragement. You were always willing to pick me up when I needed it and push me back on my way. I would also like to thank my family, Mark, Alison, Lydia, and William, for their continued support, both emotional and financial whilst I wasn't at my finest. I must also acknowledge the interest of my grandfather Michael, who ploughed through the beast with enthusiasm and absence of complaint. Finally, I thank my friends Tom and Emily, who put a roof over me in those final months and wound me down with food and drink.

AUTHOR'S DECLARATION

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: ..... DATE: .....

TABLE OF CONTENTS

List of Tables

List of Figures

Figure **Page**

CHAPTER 1

INTRODUCTION

The natural world is replete with examples of particle advection; mankind is both a beneficiary from and sufferer of the consequences. Acute economic ramifications can arise from the inability to control or predict particle behaviour. In 2010 the spread of volcanic ash cost EU businesses an estimated €2.5bn [8], and has been greatly studied as a result [9–11]. Worldwide trends of urbanisation indicate that human health will increasingly be affected by atmospheric particle pollution. In fact, in 2014 the World Health Organisation estimated that 7 million deaths can be attributed annually to air pollution [12]. Riverbed pollution is strongly affected by patterns of particulate sedimentation. As of 2015, 159 million people globally depended on untreated surface water for drinking [13]. Exploitation of particulate transport is a recent innovation in the medical field for oesophageal and blood-stream drug delivery [14]. In meteorology the dynamics of newly nucleated cloud droplets have become the subject of much research over recent years [15–17]. Regions of dense clustering in clouds enhances droplet collision rates, accelerating the growth of raindrops from nucleation diameters to precipitation. Progress in all of these fields is dependent on understanding particle and fluid interaction.

In each of the cases above, particles of one material density are suspended in a fluid of another. In such instances the particles are termed inertial and exhibit interesting behaviours. Because of the difference in their mass relative to a corresponding parcel of fluid, inertial particles do not follow fluid streamlines; the way in which they deviate from the fluid is dependent on their relative density. Aerosol particles are ejected from vortices because they are more dense than the fluid. Bubbles, which are less dense that the fluid, are attracted to the centre of vortices. This migration to and from the centre of vortices leads to preferential concentration, a phenomenon that increases particle collision rates and affects the behaviour of the fluid [16, 18]. Preferential concentration of particles is a key influence on the growth rates of droplets in warm clouds [16]. In the early stages of raindrop formation, the complex clustering and interaction properties of

the particles increase the collision and growth rates of the droplets. Raindrop formation rates are an important part of weather forecasting, but are not well understood [19].

Another area that garners significant interest, is the two-way coupling between particle and fluid. Coupling arises because the particles alter the shape of fluid streamlines, which in turn act on the particle [18, 20, 21]. Exact effects of large ensembles of particles on the fluid are not known for all cases, although there are many works that have developed models and experiments that allow investigation into these avenues [18, 20–24].

Ultimately, we study inertial particle dynamics, not because of their microscopic behaviours, but because of the range of phenomena that ensembles of them influence: from particle clustering causing accelerated rain droplet formation [25] to the presence of many small particles causing turbulence suppression [21]. Increasing our understanding of their micro-scale behaviour leads to better understanding of a wide range of macro-scale processes in the natural and engineering worlds.

One mode of investigation is experimentation. Experiments are vital for testing hypothesis, developing understanding and verifying modelling. However, it is often difficult to realise experiments that can recreate processes in the natural world. Experiments in tanks inherently have boundaries on the flow that do not reflect real world situations, and wide parameter regimes take a long time to investigate in a well resolved manner. In cases where we wish to forecast a process, we require accurate and efficient modelling. We refer to the example of ash cloud dynamics, where forecasting the particulate density and path taken by the cloud allows informed decisions to be made about the grounding of aircraft. Predictions of this type cannot be made through experiment. Not only do accurate and efficient modelling approaches for two phase flows make forecasting of particulate behaviour possible, but also models provide a versatile platform for testing hypothesis and developing understanding across wide parameter regimes. Bubbles, aerosols and neutrally buoyant particles all display different characteristics, and models provide an efficient way to understand how each of them behave in specific contexts.

Modelling inertial particles requires consideration of the influence of the fluid on a particle, the influence of particles on the fluid and the influence of particles on one another. Currently, we lack modelling approaches that satisfactorily deal with all these important dynamics of inertial particles across the space of particle parameters. In fact, it is not possible to create a model that can accurately recreate the dynamics of particles at all scales and relative densities. It is, therefore, vital to choose an appropriate model for the circumstances one wishes to describe.

Often, a wide range of scales are important in problems involving inertial particles. In many of the contexts we have cited (e.g. volcanic ash cloud dispersion), the fluid flow fields are turbulent and have a continuous range of important length scales with vortical structures at each of them [26]. Important scales range from the scales of the largest eddies to the Kolmogorov length scale, where energy is dissipated into heat [26]. Sometimes it is possible to numerically simulate all the length scales and to include the presence of particles in the discretisation.

However, when particles are numerous, attempting to represent each of them in a discretisation is computationally expensive. Some of the largest simulations that model inertial particles as fully resolved objects in a flow have only achieved $O(1000)$ particles [24, 27, 28]. Processes in the natural world which we wish to model, contain many more particles in much smaller regions of the flow.

Let us take the example of cloud microphysics for which we want to create a numerical model. First we must consider a reasonable domain size for discretisation. Despite the constant improvement in computing power, all the important length scales cannot be resolved. The largest scales in a cloud are of the order of 1km, whilst the newly nucleated droplets are of the order of 1μm [29]. Such a breadth of scales, covering 9 orders of magnitude, is far from discretisable with one model. However, it is of little concern to the dynamics of the particles how the fluid velocity is behaving at the large scales of a cloud. The important concern for the particles are the small scales close to their own diameter. A discretisable domain can be obtained using periodic boundary conditions and assuming that a small subsection of a cloud flow is repeated homogeneously throughout the cloud. However, continental clouds can contain 1000 particles per cubic centimetre [29]. For a simulation with a periodic domain representing 1cm^{-3} of cloud, the number of particles in the simulation is close to the largest simulations of fully resolved spheres in a flow. Clearly, modelling particles as fully resolved objects in a flow is not suitable for all applications.

Another consideration for the application of raindrop formation is the interaction between particles. In warm clouds, particle collisions are of particular concern [25]. Collision detection algorithms scale poorly with the number of particles, $O(N^{3/2})$, which dramatically increases the cost of simulating the process numerically.

In other applications where the fluid velocity field is turbulent, such as sediment transportation or planetary accretion, predicting trajectories of individual particles is extremely challenging. Small errors in the calculations of trajectories or poor choice of model, compound to produce large errors in the final positions of the particles. However, in these cases one is not particularly interested in the path taken by a single grain of sand, rather it is important to understand how the whole sand ensemble behaves. For modelling a large number of grains of sand an Eulerian approach may be more suitable.

Evidently, appropriate model choice for a specific application is vital. Appropriate modelling is largely dependent on the scales of the particles relative to the scales of the flow. For very small particles, many orders of magnitude smaller than the Kolmogorov length scale, model choice is trivial because particles follow fluid streamlines exactly. Large particles, many times larger than the Kolmogorov length scale, disturb the fluid sufficiently to require particles to be fully resolved objects in a flow. One example of this being the "direct forcing method" [27, 28]. In between, it is not always clear which phenomena should be included to achieve accurate models. Particularly uncertain are the circumstances that require the use of a force known as the Basset history force

[30–32].

We aim to improve clarity in appropriate model choice, and develop modelling approaches that allow particle processes to be described more fully. In this thesis two problems are considered in detail: firstly the sedimentation of inertial particles for application in sediment transport and seabed dredging, and secondly the early stages of raindrop formation where collisions between particles are of particular concern. Accurate modelling of these processes requires consideration of the two-way coupling between the particles and the fluid, as well as the interaction of particles with one another. We present a series of novel modelling approaches that allow these interactions to be investigated in a computationally efficient manner. In particular we have developed techniques that improve the scaling performance of collision detection. Throughout this thesis, particles are assumed to be spherical and the terms 'particle' and 'sphere' are used interchangeably.

We begin with a discussion of the details of fluid-particle interaction, starting from the simple case of a single spherical particle suspended in a fluid flow. The exact nature of the force contributions imposed on a sphere depend on the fluid velocity relative to it: the combination of fluid viscosity, relative velocity and sphere radius, change the shape of fluid streamlines around the sphere.

A key parameter that indicates the type of flow around the sphere is the dimensionless particle Reynolds number, Re_p. Consider a sphere with velocity $\mathbf{V}(t)$, traveling in a fluid with local velocity $\mathbf{u}(\mathbf{x},t)$, the particle Reynolds number, Re_p, is given by,

$$Re_p = \frac{2\rho_f r_p(\mathbf{V}(t)-\mathbf{u}(\mathbf{x},t))}{\mu}, \tag{1.1}$$

where ρ_f is the density of the fluid, μ is the dynamic viscosity of the fluid and r_p is the particle radius. The particle Reynolds number is a ratio of the inertial forces to the viscous forces acting on the sphere. Figure 1.1 shows how the flow around a sphere changes for increasing particle Reynolds number. When the relative velocities between the sphere and the fluid are small, the flow around the object is predictable and laminar, and the dominant force is viscous drag [1]. However, when the relative velocity is large, the flow separates from the sphere entirely and a turbulent wake forms behind the sphere. The fluid pressure is reduced in the turbulent wake and the dominant force acting on the sphere is pressure drag.

Other forces play roles in certain circumstances. Asymmetry in the flow incident to the sphere produces the Saffman lift force [33], and the Magnus lift force is caused by rotation of the particle [34]. A force called the added mass force arises due to the fluid that a sphere must move in order to make progress [1]. Oscillatory motion in a particle trajectory can be caused by the vortices shed behind the particle [35]. When multiple spheres are present in a flow, the flows around each sphere not only have a bearing on the trajectory of that sphere, but on the trajectory of all other spheres in the flow.

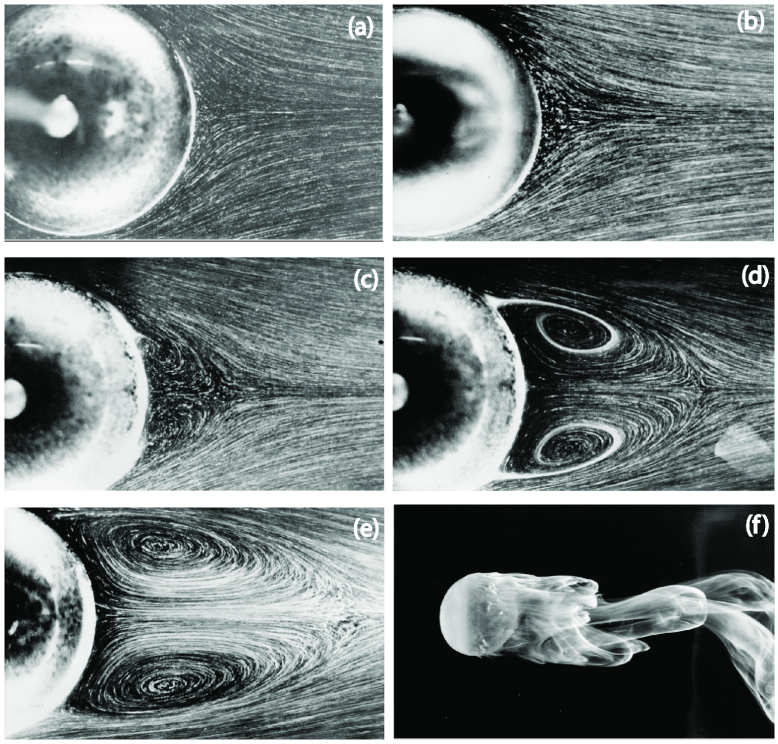


Figure 1.1: *A visualisation of the flow around a sphere is shown for increasing Re_p. As the Reynolds number increases, vortical structures develop in the wake. a)$Re_p = 9.15$; (b) $Re_p = 17.9$; (c) $Re_p = 37.7$; (d) $Re_p = 73.6$; (e) $Re_p = 118$; and (f) $Re_p = 550$. The figure was taken from [1] but originally produced by [2]*

To describe all the cases in figure 1.1 accurately, the Navier-Stokes equations must be solved around the sphere. The incompressible Navier-Stokes equations are given by,

$$\rho\left(\frac{\partial \mathbf{u}(\mathbf{x},t)}{\partial t} + (\mathbf{u}(\mathbf{x},t)\cdot\nabla)\mathbf{u}(\mathbf{x},t)\right) = -\nabla p + \mu\nabla^2\mathbf{u}(\mathbf{x},t) + \rho g \qquad (a)$$
$$\nabla\cdot\mathbf{u}(\mathbf{x},t) = 0 \qquad (b) \tag{1.2}$$

where, $\mathbf{u}(\mathbf{x},t)$ is the fluid velocity, μ is the dynamic viscosity of the flow, p is the fluid pressure, g is acceleration due to gravity and ρ is the density of the fluid. The forces acting on a sphere can be found by considering how the fluid stresses are distributed over its surface.

Consider a rigid spherical particle advected by the fluid velocity field $\mathbf{u}(\mathbf{x},t)$. The equation of motion for a particle of mass m_p, with position $Y_i(t)$ and travelling with velocity $V_i(t)$ in the flow $\mathbf{u}(\mathbf{x},t)$, is found by considering Newton's second law for the particle,

$$m_p\frac{dV_i}{dt} = \sum_j F_i^j, \tag{1.3}$$

where $\sum_j F_i^j$, is the sum of the forces acting on the particle: buoyancy force and the contributions from the fluid. In this thesis we use Einstein's summation notation. For convenience we also relax the notation that, u_i, $V_i(t)$ and $Y_i(t)$,must be written as functions of z_i and t. From here v_i implies $\mathbf{v}(\mathbf{x},t)$, similarly u_i implies $\mathbf{u}(\mathbf{x},t)$ and V_i and Y_i imply $V_i(t)$ and $Y_i(t)$ respectively. The contribution from the fluid can be found by considering how the fluid stresses are distributed over the sphere. The fluid stress tensor, σ_{ij}, is given by,

$$\sigma_{ij} = -p\delta_{ij} + \mu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right), \tag{1.4}$$

where δ_{ij} is the Kronecker-delta. To find the force contributions from the fluid stresses, we integrate $\sigma_{ij}n_j$ over the surface of the sphere. The equation of motion for the particle is equal to the buoyancy force and the integral of the fluid stresses over the sphere's surface,

$$m_p\frac{dV_i}{dt} = m_p g_i + \oiint_S \sigma_{ij}\cdot n_j dS, \tag{1.5}$$

where n_j is the outward unit vector, normal to the surface of the sphere.

In general, it is not possible to integrate the stress tensor around the sphere analytically. Fortunately, there are numerical approaches that can calculate fluid forces acting on a sphere. Methods such as immersed boundary methods [36] and smoothed particle hydrodynamics [37], are good candidates for numerical models of multiphase flows [28]. However, as with analytic solutions, no numerical method will solve our problem for all possible cases.

As discussed previously, the range of important length scales can be very large. In turbulent flows particularly, important scales have a wide distribution with no separation between them [26]. It is often not possible to discretise a space that covers the largest scales, whilst simultaneously resolving the smallest. If however, there is a separation between important scales, separate models can be used for the large scale and small scale behaviour. In this thesis, we exploit a separation of scales between the particles and the fluid velocity field. Whilst the smallest scales in turbulent flows have the greatest effect on the dynamics of inertial particles [7], it is generally considered that fluid velocity variations are insignificant if they are smaller than the Kolmogorov length scale [26]. Therefore, in cases where particles are smaller than the Kolmogorov length scale of a flow, we are able to use one model for the behaviour of the particles, and one for the fluid.

When particles are small, the flow around them is laminar and a sufficiently good approximation to the flow around the sphere can be found analytically [38]. In such cases, we can integrate analytically the fluid stresses over the sphere to find the sphere's equation of motion. A large body of literature has developed low Reynolds number approximations to equations of motion for spheres. In the following section we present a brief discussion of the history of this body of literature.

The first attempts at low Reynolds number approximations to the description of a rigid sphere travelling in a fluid came in the 1800s. At the time the work was motivated by improving clock

accuracy [39–41]. As a clock pendulum follows a sinusoidal path, the early researchers calculated the forces acting on the sphere for the specific case of sinusoidal motion.

The first equation of motion for a sphere in a fluid was found in 1885 by Boussinesq [42]. His equation described the unsteady motion of a sphere in a fluid, and was repeated independently by Basset a few years later [43]. Their derivations were made under the assumption that the flow around the sphere is laminar and resembles the flow in figure 1.1(a). In the early 20th century, Oseen worked on improving the drag coefficient for non-zero Reynolds numbers [44, 45].

The early works of Basset, Boussinesq and Oseen, led to the equation of motion for a small rigid spherical particle in a fluid being termed the Basset-Boussinesq-Oseen equation. A number of subsequent improvements and extensions have been made to their original equations of motion. A review by Michaelides gives a detailed account of the history of the Basset-Bousinesq-Oseen equation (BBO equation) [46]. We focus here on an important contribution from Maxey and Riley, whose equation of motion is known as the Maxey-Riley equation [47]. Their contribution was to present a derivation with the limitations, complexities and assumptions clearly stated.

In order to understand assumptions they make, we present in the following section Maxey and Riley's derivation, which is the starting point for much of our later modelling. Maxey and Riley assume that the fluid velocity field can be described by a linear superposition of two flows: the first flow is the fluid velocity field that would exist in the absence of the particle, which we call $\mathbf{u}(\mathbf{x},t)$; the second flow, which we call the disturbance, is the change in $\mathbf{u}(\mathbf{x},t)$ caused by the presence of the particle. Superimposing the disturbance and the undisturbed fluid velocity leads to the total fluid velocity, $\mathbf{v}(\mathbf{x},t)$. We recall that the forces acting on the sphere are given by the integral of the fluid stresses over the particle's surface,

$$
\begin{aligned}
m_p \frac{dV_i}{dt} &= m_p g_i + \oiint_S \sigma_{ij} n_j dS, \quad \text{(a)} \\
\sigma_{ij} &= -p\delta_{ij} + \mu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right), \quad \text{(b)}
\end{aligned}
\tag{1.6}
$$

where v_i is the disturbed flow around the particle. For a sphere of radius r_p, at position $Y(t)$, travelling with velocity $V(t)$, we assume that v_i is governed by the following equations,

$$
\begin{aligned}
&\rho\left(\frac{\partial v_i}{\partial t} + v_j\frac{\partial v_i}{\partial x_j}\right) = -\frac{\partial p}{\partial x_i} + \mu\frac{\partial^2 v_i}{\partial x_j \partial x_j} + \rho g_i && (a) \\
&\frac{\partial v_i}{\partial x_i} = 0 && (b) \\
&\mathbf{v}(\mathbf{x},t) = \mathbf{V}(t) + \Omega \times [\mathbf{x} - \mathbf{Y}(t)] \text{ for } |\mathbf{x} - \mathbf{Y}(t)| = r_p && (c) \\
&v_i = u_i \text{ as } |x_i - Y_i(t)| \to \infty && (d)
\end{aligned}
\tag{1.7}
$$

where Ω is the angular velocity of the spherical particle and $\times$ denotes the cross product. The position of a fluid element is given by $\mathbf{x}$, p is the fluid pressure, μ and ρ_f are the dynamic viscosity and density of the fluid respectively and g is acceleration due to gravity. Equation

1.7(a) is the incompressible Navier-Stokes momentum equation, with the associated statement of incompressibility given in (b). Equation 1.7(c) ensures that, on the surface of the sphere, the fluid velocity $\mathbf{v}(\mathbf{x},t)$ is equal to the sphere's velocity. The final condition, (d), assumes that the disturbance caused by the particle decays to zero far from the sphere: $\mathbf{v}(\mathbf{x},t)$, is equal to the undisturbed background flow as the distance from the sphere tends to infinity.

We make a change of coordinates to a reference frame moving with the centre of the sphere given by $\mathbf{z} = \mathbf{x} - \mathbf{Y}(t)$, and $\mathbf{w}(\mathbf{z},t) = \mathbf{v}(\mathbf{z},t) - \mathbf{V}(t)$. The conditions in equation 1.7 become,

$$
\begin{aligned}
&\rho\left(\frac{\partial w_i}{\partial t} + w_j\frac{\partial w_i}{\partial z_j}\right) = -\frac{\partial p}{\partial z_i} + \mu\frac{\partial^2 w_i}{\partial z_j \partial z_j} + \rho\left(g - \frac{dV_i}{dt}\right) && (a)\\
&\frac{\partial w_j}{\partial z_j} = 0 && (b)\\
&\mathbf{w}(\mathbf{z},t) = \Omega \times \mathbf{z} \text{ for } \mathbf{z} = r_p && (c)\\
&w_i = u_i - V_i \text{ as } |z| \to \infty && (d).
\end{aligned}
\tag{1.8}
$$

Maxey and Riley exploit a separation of scales between the disturbance caused by the particle and the undisturbed fluid velocity field. By assuming a separation of scales, we can describe $\mathbf{w}(\mathbf{z},t)$ as a linear combination of two flows, such that, $\mathbf{w}(\mathbf{z},t) = \alpha(\mathbf{z},t) + \beta(\mathbf{z},t)$. The two flows are the disturbance caused by the particle, $\beta(\mathbf{z},t)$, and the fluid velocity field in the absence of the particle, $\alpha(\mathbf{z},t)$. We will refer to $\alpha(\mathbf{z},t)$ as the background flow, an arbitrary flow which satisfies the incompressible Navier-Stokes equations. It can be related to the original coordinate frame as $\alpha(\mathbf{z},t) = \mathbf{u}(\mathbf{z},t) - \mathbf{V}(t)$. We will refer to $\beta(\mathbf{z},t)$ as the disturbance, which is the velocity field resulting only from the presence of the particle, and is independent of $\alpha(\mathbf{z},t)$. A separate pressure is also attributed to each flow, p^α to the background flow and p^β to the disturbance flow, so that $p^\alpha + p^\beta = p$.

The fluid velocity in its two component parts must satisfy,

$$
\begin{aligned}
&\rho\left(\frac{\partial \alpha_i}{\partial t} + \frac{\partial \beta_i}{\partial t} + \beta_j\frac{\partial \beta_i}{\partial z_j} + \alpha_j\frac{\partial \beta_i}{\partial z_j} + \alpha_j\frac{\partial \alpha_i}{\partial z_j} + \beta_j\frac{\partial \alpha_i}{\partial z_j}\right) =\\
&\qquad \mu\frac{\partial^2 \alpha_i}{\partial z_j \partial z_j} + \mu\frac{\partial^2 \beta_i}{\partial z_j \partial z_j} - \frac{\partial (p^\alpha + p^\beta)}{\partial z_i} + \rho\left(g - \frac{dV_i}{dt}\right).
\end{aligned}
\tag{1.9}
$$

and must also independently satisfy the Navier-Stokes equations: equation 1.9 can be separated

into the following two conditions,

$$
\begin{aligned}
&\rho\left(\frac{\partial \alpha_i}{\partial t}+\alpha_j\frac{\partial \alpha_i}{\partial z_j}\right)=\mu\frac{\partial^2\alpha_i}{\partial z_j\partial z_j}-\frac{\partial p^\alpha}{\partial z_i}+\rho\left(g-\frac{dV_i}{dt}\right) && (a),\\
&\frac{\partial \alpha_j}{\partial z_j}=0 && (b)\\
&\rho\left(\frac{\partial \beta_i}{\partial t}+\beta_j\frac{\partial \beta_i}{\partial z_j}+\alpha_j\frac{\partial \beta_i}{\partial z_j}+\beta_i\frac{\partial \alpha_i}{\partial z_j}\right)=\mu\frac{\partial^2\beta_i}{\partial z_j\partial z_j}-\frac{\partial p^\beta}{\partial z_i} && (c)\\
&\frac{\partial \beta_j}{\partial z_j}=0 && (d)
\end{aligned}
\tag{1.10}
$$

The force acting on the sphere from the background flow, F_i^α, can be found by integrating the fluid stress tensor for the background flow over the surface of the sphere. Through the divergence theorem the surface integral can be transformed into a volume integral,

$$
\begin{aligned}
&F_i^\alpha=\oiint_S \sigma_{ij}n_j dS=\iiint_V\frac{\partial}{\partial z_j}\sigma_{ij}dV=\\
&\int_0^\pi\int_0^{2\pi}\int_0^{r_p}\left\{\frac{\partial}{\partial z_j}\left(\mu\left(\frac{\partial \alpha_i}{\partial z_j}+\frac{\partial \alpha_j}{\partial z_i}\right)\right)-\frac{\partial p^\alpha}{\partial z_i}\right\}r^2\sin(\phi)dr d\theta\, d\phi
\end{aligned}
\tag{1.11}
$$

By expanding the integrand, $\nabla\cdot\sigma$, into its components $\alpha_i=[u,v,w]$ and $z_i=[x,y,z]$ we can see that it can be simplified through the condition of incompressibility,

$$
\frac{\partial}{\partial z_j}\sigma_{ij}=\mu\begin{pmatrix}\frac{\partial^2u}{\partial x^2}+\frac{\partial^2u}{\partial y^2}+\frac{\partial^2u}{\partial z^2}+\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\\ \frac{\partial^2v}{\partial x^2}+\frac{\partial^2v}{\partial y^2}+\frac{\partial^2v}{\partial z^2}+\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)\\ \frac{\partial^2w}{\partial x^2}+\frac{\partial^2w}{\partial y^2}+\frac{\partial^2w}{\partial z^2}+\frac{\partial}{\partial z}\underbrace{\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}\right)}_{\nabla\cdot\vec{\mathbf{v}}=0}\end{pmatrix}-\begin{pmatrix}\frac{\partial p^\alpha}{\partial x}\\ \frac{\partial p^0}{\partial \tilde{y}}\\ \frac{\partial p^0}{\partial \tilde{z}}\end{pmatrix}=\mu\left(\frac{\partial^2\alpha_i}{\partial z_j\partial z_j}\right)-\frac{\partial p^\alpha}{\partial z_i},
\tag{1.12}
$$

and we are left to evaluate,

$$
F_i^\alpha=\oiint_S\sigma_{ij}\cdot n_j dS=\int_0^\pi\int_0^{2\pi}\int_0^{r_p}\left\{\mu\left(\frac{\partial^2\alpha_i}{\partial z_j\partial z_j}\right)-\frac{\partial p^\alpha}{\partial z_i}\right\}r^2\sin(\phi)dr d\theta d\phi.
\tag{1.13}
$$

As we know a separation of scales is assumed between the length scales of the background flow, and the scale of the particle. By assuming that over the scale of the sphere, the pressure gradient and spatial velocity derivatives are constant, they can be moved outside the integrand. Evaluation of the volume integral gives,

$$
F_i^\alpha=\frac{4}{3}\pi r_p^3\left(\mu\left(\frac{\partial^2\alpha_i}{\partial z_j\partial z_j}\right)-\frac{\partial p^\alpha}{\partial z_j}\right),
\tag{1.14}
$$

which are the forces acting on the sphere from the undisturbed portion of the flow. Combining equations (1.10) and (1.14), gives the first three terms of the MR equation,

$$
F_i^\alpha=\frac{4}{3}\pi\rho r_p^3\left(\left(\frac{\partial \alpha_i}{\partial t}+\alpha_j\frac{\partial \alpha_i}{\partial z_j}\right)-\left(g-\frac{dV_i}{dt}\right)\right)=m_f\frac{D\alpha_i}{Dt}-m_f g+m_f\frac{dV_i}{dt},
\tag{1.15}
$$

where m_f is the mass of a fluid parcel equal in volume to the sphere. Returning to the original coordinate frame the equation of motion for the sphere can be written down as,

$$m_p \frac{d\mathbf{V}}{dt} = m_f \frac{D\mathbf{u}}{Dt} + (m_p - m_f)\mathbf{g} + \sum_j F_j, \tag{1.16}$$

where m_p is the mass of the sphere. We still require expressions for $\sum_j F_j$, which are the forces imposed on the sphere by the disturbance flow, $\beta(\mathbf{z},t)$.

We now turn to the disturbance flow $\beta(\mathbf{z},t)$. As discussed, we are assuming that the flow around the sphere is laminar. A laminar flow can be described by linear equations in which the non-linear advective terms from the full Navier-Stokes are assumed negligible and ignored. The resulting equations are known as Stokes equation and describe an incompressible Stokes flow. Stokes momentum equation and the Navier-Stokes momentum equation, are shown side by side in equation 1.17. The terms that have been neglected are indicated.

$$\begin{aligned} \rho\left(\frac{\partial \mathbf{u}(\mathbf{x},t)}{\partial t} + \underbrace{(\mathbf{u}(\mathbf{x},t)\cdot\nabla)\mathbf{u}(\mathbf{x},t)}_{\text{advective terms}}\right) &= -\nabla p + \mu\nabla^2\mathbf{u}(\mathbf{x},t) + \rho g \qquad \text{(Navier-Stokes momentum)} \\ \rho\frac{\partial \mathbf{u}(\mathbf{x},t)}{\partial t} &= -\nabla p + \mu\nabla^2\mathbf{u}(\mathbf{x},t) + \rho g \qquad \text{(Stokes flow momentum)} \end{aligned} \tag{1.17}$$

The disturbance flow must satisfy the boundary conditions,

$$\begin{aligned} &\beta = (\mathbf{V} - \mathbf{u}) + \Omega \times \mathbf{z} \quad \text{for} \quad \mathbf{z} = r_p \\ &\beta \to 0 \quad \text{as } |\mathbf{z}| \to \infty \end{aligned} \tag{1.18}$$

For the disturbance flow the non-linear terms from equation 1.10(b) are neglected and Lorenz reciprocal theorem is employed. With non-linear terms removed, we define two unsteady Stokes flows, $\mathbf{v}(\mathbf{z},t)$ and $\mathbf{v}'(\mathbf{z},t)$, in some arbitrary volume $\widehat{V}$ bounded by a surface $\widehat{S}$, that must satisfy

$$\begin{aligned} \rho\frac{\partial v_i}{\partial t} - \frac{\partial}{\partial x_j}\sigma_{ij} = 0 \quad \text{(a)} \\ \frac{\partial v_i}{\partial x_i} = 0 \quad \text{(b).} \end{aligned} \tag{1.19}$$

For the Stokes flows in equation 1.19(a) and (b), the Lorenz reciprocal theorem states that,

$$\oiint_{\widehat{S}} v_i \sigma'_{ij} n_j d\widehat{S} = \oiint_{\widehat{S}} v'_i \sigma_{ij} n_j d\widehat{S}. \tag{1.20}$$

Through the divergence theorem we find,

$$\iiint_{\widehat{V}} \left(v_i \frac{\partial}{\partial z_j}\sigma'_{ij} + \sigma'_{ij}\frac{\partial}{\partial z_j} v_i\right) d\widehat{V} = \oiint_{\widehat{S}} v_i \sigma'_{ij} n_j d\widehat{S} \tag{1.21}$$

and similarly,

$$\iiint_{\widehat{V}} \left(v_i' \frac{\partial}{\partial z_j} \sigma_{ij} + \sigma_{ij} \frac{\partial}{\partial z_j} v_i' \right) d\widehat{V} = \oiint_{\widehat{S}} v_i' \sigma_{ij} n_j d\widehat{S}. \tag{1.22}$$

We now take 1.22 from 1.21 to give,

$$\iiint_{\widehat{V}} \left(v_i \frac{\partial}{\partial z_j} \sigma_{ij}' + \sigma_{ij}' \frac{\partial}{\partial z_j} v_i - v_i' \frac{\partial}{\partial z_j} \sigma_{ij} - \sigma_{ij} \frac{\partial}{\partial z_j} v_i' \right) d\widehat{V} = \oiint_{\widehat{S}} \left(v_i \sigma_{ij}' - v_i' \sigma_{ij} \right) n_j d\widehat{S}. \tag{1.23}$$

Through analysis of individual components it is possible to show that,

$$\sigma_{ij}' \frac{\partial}{\partial z_j} v_i = \sigma_{ij} \frac{\partial}{\partial z_j} v_i' \tag{1.24}$$

which implies that,

$$\iiint_{\widehat{V}} \left(v_i \frac{\partial}{\partial z_j} \sigma_{ij}' - v_i' \frac{\partial}{\partial z_j} \sigma_{ij} \right) d\widehat{V} = \oiint_{\widehat{S}} \left(v_i \sigma_{ij}' - v_i' \sigma_{ij} \right) n_j d\widehat{S}. \tag{1.25}$$

Now, substituting equation 1.19a into equation 1.25 gives,

$$\iiint_{\widehat{V}} \rho \left(v_i \frac{\partial}{\partial t} v_i' - v_i' \frac{\partial}{\partial t} v_i \right) d\widehat{V} = \oiint_{\widehat{S}} \left(v_i \sigma_{ij}' - v_i' \sigma_{ij} \right) n_j d\widehat{S}, \tag{1.26}$$

Next we take the Laplace transform of equation 1.26. For the Laplace transform of $\mathbf{v}(\mathbf{x},t)$, given by $\mathscr{L}[\mathbf{v}(\mathbf{x},t)] = \int_0^\infty \mathbf{v}(\mathbf{x},t) e^{-st} dt$, we use the notation $\widetilde{\mathbf{v}}(\mathbf{x},s) = \mathscr{L}[\mathbf{v}(\mathbf{x},t)]$.

$$\iiint_{\widehat{V}} \rho \left(\widetilde{v}_i \left[s\widetilde{v}_i' - v_i'(x,0) \right] - \widetilde{v}_i' \left[s\widetilde{v}_i - v_i(x,0) \right] \right) d\widehat{V} = \oiint_{\widehat{S}} \left(\widetilde{v}_i \widetilde{\sigma}_{ij}' - \widetilde{v}_i' \widetilde{\sigma}_{ij} \right) n_j d\widehat{S} \tag{1.27}$$

which simplifies to,

$$\iiint_{\widehat{V}} \rho \left(v_i(x,0) \widetilde{v}_i' - v_i'(x,0) \widetilde{v}_i \right) d\widehat{V} = \oiint_{\widehat{S}} \left(\widetilde{v}_i \widetilde{\sigma}_{ij}' - \widetilde{v}_i' \widetilde{\sigma}_{ij} \right) n_j d\widehat{S} \tag{1.28}$$

The flows must satisfy the following boundary conditions,

$$\begin{aligned} &|\mathbf{v}| \longrightarrow 0 \text{ and } |\mathbf{v}'| \longrightarrow 0 \text{ as } |\mathbf{x}| \longrightarrow \infty \\ &\widetilde{\mathbf{v}}'(r_p, s) = (1,0,0) = e_1 \text{ where } r_p \text{ is the radius of the sphere,} \\ &\mathbf{v}'(x, t \le 0) = 0 \end{aligned} \tag{1.29}$$

The disturbance flows v_i and v_i' represent the part of the flow caused by the spheres presence and so decay to zero far from the sphere. Therefore, only integrals over the surface of the sphere are important. To find the force acting on the sphere we consider that the sphere starts from rest at time $t = 0$ and then undergoes an impulsive velocity $[\delta(t), 0, 0]$. In the Laplace domain the

impulsive velocity is given by $\mathscr{L}[\delta(t),0,0] = [1,0,0]$. Substitution of the boundary conditions in equation 1.29 into equation 1.26 leads to an expression for the force acting on the sphere,

$$\begin{aligned}\widetilde{F}_1 &= \oiint_{\widehat{S}} e_i(1)\widetilde{\sigma}_{ij}n_j d\widehat{S} = \oiint_{S} \widetilde{v}_i\widetilde{\sigma}'_{ij}n_j d\widehat{S} - \iiint_{\widehat{V}} \rho \quad \left(\underbrace{v_i(x,0)\widetilde{v}'_i}_{0} - \underbrace{v'_i(x,0)\widetilde{v}_i}_{=0}\right) d\widehat{V} \\ \widetilde{F}_1 &= \oiint_{\widehat{S}} \widetilde{v}_i\widetilde{\sigma}'_{ij}n_j d\widehat{S} - \underbrace{\iiint_{\widehat{V}} \rho v_i(x,0)e_i^{(1)} d\widehat{V}}_{0}.\end{aligned} \tag{1.30}$$

where $e_i^{(1)} = (1,0,0)$. The Stokes flow around a sphere is given by Burgers solution [38],

$$\begin{aligned}\widetilde{v}'_i &= \left(e_i^{(1)}\frac{\partial}{\partial x_i}\right)\frac{\partial\widetilde{\Psi}}{\partial x_i} - e_i^{(1)}\frac{\partial^2\widetilde{\Psi}}{\partial x_i\partial x_i} \\ \widetilde{\Psi}(r,s) &= Q_1 sr\rho + Q_2\exp(-\lambda r)r\end{aligned} \tag{1.31}$$

where $\lambda = \sqrt{s\rho_f/\mu}$, ρ_f is the fluid density, $r = \sqrt{x^2+y^2+z^2}$ and μ is the dynamic viscosity of the fluid. We use Burgers solution for the flow $\widetilde{v}'_i$, from which it is possible to calculate $\widetilde{\sigma}'_{ij}$,

$$\widetilde{\sigma}'_{ij}n_j = -\frac{\rho_f s x_i x_j e_j^{(1)}}{2r_p} - \frac{3\mu(1+\lambda r_p)e_i^{(1)}}{r_p}. \tag{1.32}$$

On the sphere, Maxey and Riley assume that $\widetilde{v}_i$ has the form,

$$\widetilde{v}_i = \widetilde{A}_i + x_j\widetilde{B}_{ij} + \frac{1}{2}x_j x_k\widetilde{C}_{ijk} + h.o.t. \tag{1.33}$$

Further, they assume that $\widetilde{v}_i(x_i,0) = 0$, allowing us to find the required product in the only remaining integral of equation 1.30. It is now possible to find F_i by conducting a series of surface integrals over the surface of the sphere. First we consider the terms multiplied by $\widetilde{A}_i$ which are given by,

$$\begin{aligned}&-\frac{\widetilde{A}_i}{2r_p}\oiint_{\widehat{S}} \rho s x_i x_j e_j^{(1)} d\widehat{S}, \quad (a) \\ &-\frac{\widetilde{A}_i}{2r_p}\oiint_{\widehat{S}} 3\mu(1+\lambda r_p)e_i^{(1)} d\widehat{S} \quad (b).\end{aligned} \tag{1.34}$$

Expansion of the integrand in equation 1.34(a) gives $\widetilde{A}_1x^2 + \widetilde{A}_2xy + \widetilde{A}_3xz$, which in spherical

polar coordinates gives,

(1.35)

$$
\begin{aligned}
&-\frac{\rho s}{2r_p}\int_0^{2\pi}\int_0^{\pi}\left(\widetilde{A}_1 r_p^2\sin^2(\phi)cos^2(\theta)+\widetilde{A}_2 r_p^2\sin^2(\phi)\cos(\theta)\sin(\theta)+\widetilde{A}_3 r_p^2\sin\phi\cos\theta\cos\phi\right)r_p^2\sin(\phi)d\phi d\theta\\
&=-\frac{\rho s r_p^3}{2}\int_0^{2\pi}\int_0^{\pi}\left(\widetilde{A}_1\sin^3(\phi)cos^2(\theta)+\widetilde{A}_2\sin^3(\phi)\cos(\theta)\sin(\theta)+\widetilde{A}_3\sin^2\phi\cos\theta\cos\phi\right)d\phi d\theta\\
&=-\frac{\rho s r_p^3}{2}\int_0^{2\pi}\left[\frac{1}{12}\left(\cos(3\phi)-9\cos(\phi)\right)(\widetilde{A}_1\cos^2\theta+\widetilde{A}_2\sin(\theta)\cos(\theta))+\widetilde{A}_3\frac{\sin^3\phi}{3}\cos(\theta)\right]_0^{\pi}d\theta\\
&=-\frac{\rho s r_p^3}{2}\int_0^{2\pi}\frac{4}{3}(\widetilde{A}_1\cos^2(\theta)+\widetilde{A}_2\cos(\theta)\sin(\theta))d\theta\\
&=-\frac{\rho s r_p^3}{2}\frac{4}{3}\left[\frac{1}{2}(\widetilde{A}_1(\theta+\sin(\theta)\cos(\theta))-\widetilde{A}_2\cos^2(\theta))\right]_0^{2\pi}\\
&=-\widetilde{A}_1\frac{1}{2}\frac{4\pi\rho s r_p^3}{3}.
\end{aligned}
$$

The integrand in equation 1.34(b) is a constant. We know that the surface area of a sphere is $4\pi r_p^2$ so,

$$
-\frac{\widetilde{A}_i}{2r_p}\oiint_{\widehat{S}}(3\mu(1+\lambda r_p))d\widehat{S}=-A_1 6\pi r_p\mu(1+\lambda r_p) \tag{1.36}
$$

All integrals involving $\widetilde{B}_{ij}$ are equal to zero and we are left with the $\widetilde{C}_{ijk}$ terms,

$$
-\frac{1}{2r_p}\oiint_{\widehat{S}}(\underbrace{(x_j x_k C_{ijk})\rho s x_i x_j e_j^{(1)}}_{\text{integral 1}}+\underbrace{(3x_j x_k C_{ijk})\mu(1+\lambda r_p)e_i^{(1)}}_{\text{integral 2}})d\widehat{S} \tag{1.37}
$$

With help from some computer algebra we arrive at,

$$
-r_p^3\pi\mu(1+\lambda r_p)(\widetilde{C}_{1jj})-\frac{r_p^2 s\rho\pi}{15}(3\widetilde{C}_{111}+\widetilde{C}_{122}+\widetilde{C}_{133}+\widetilde{C}_{212}+\widetilde{C}_{313}+\widetilde{C}_{221}+\widetilde{C}_{331}) \tag{1.38}
$$

and combining the terms containing $\widetilde{A}_i$ and $\widetilde{C}_{ijk}$ we reach an expression for the force acting on the sphere in the first dimension,

$$
\begin{aligned}
\widetilde{F}_1=&-\widetilde{A}_1\frac{1}{2}sm_f-\widetilde{A}_1 6\pi r_p\mu(1+\lambda rP)\\
&-r_p^3\pi\mu(1+\lambda r_p)(\widetilde{C}_{1jj})\\
&-\frac{r_p^2 s m_f}{20}(\widetilde{C}_{1jj}+\widetilde{C}_{j1j}+\widetilde{C}_{jj1})
\end{aligned} \tag{1.39}
$$

where m_f is the mass of a parcel of fluid equal in volume to the volume of the sphere. In Maxey and Riley's paper there appears to be a small error at this point. Instead of the expression in

equation 1.39, they obtain,

$$\begin{aligned}\widetilde{F}_1 = &-\widetilde{A}_1\frac{1}{2}sm_f - \widetilde{A}_1 6\pi r_p\mu(1+\lambda rP)\\ &-r_p^3\pi\mu(1+\lambda r_p)(\widetilde{C}_{1jj}+2\widetilde{C}_{jj1})\\ &-\frac{r_p^2 sm_f}{20}(\widetilde{C}_{1jj}),\end{aligned} \tag{1.40}$$

which does not correspond to the subsequent results in their paper. By repeating the steps above for each dimension we obtain the correct expression for the 3-dimensional force acting on the sphere,

$$\begin{aligned}\widetilde{F}_i = &-\widetilde{A}_i\frac{1}{2}sm_f - \widetilde{A}_i 6\pi r_p\mu(1+\lambda r_p)\\ &-r_p^3\pi\mu(1+\lambda r_p)(\widetilde{C}_{ijj})\\ &-\frac{r_p^2 sm_f}{20}(\widetilde{C}_{ijj}+\widetilde{C}_{jij}+\widetilde{C}_{jji})\end{aligned} \tag{1.41}$$

By comparing the boundary conditions in equation 1.18(a) with equation 1.33, we can see that $A_i = V_i - u_i$ and by extension, $C_{ijk} = \frac{\partial^2 u_i}{\partial x_j \partial x_k}$. Substituting these boundary conditions into equation 1.41, leads to,

$$\begin{aligned}\widetilde{F}_i = &-(\widetilde{V}_i-\widetilde{u}_i)\frac{1}{2}sm_f - (\widetilde{V}_i-\widetilde{u}_i)6\pi r_p\mu(1+\lambda r_p)\\ &-r_p^3\pi\mu(1+\lambda r_p)\frac{\partial^2\widetilde{u}_i}{\partial x_j\partial x_j}\\ &-\frac{r_p^2 sm_f}{20}\left(\frac{\partial^2\widetilde{u}_i}{\partial x_j\partial x_j}+\frac{\partial^2\widetilde{u}_j}{\partial x_i\partial x_j}+\frac{\partial^2\widetilde{u}_j}{\partial x_j\partial x_i}\right),\end{aligned} \tag{1.42}$$

the final term of which can be simplified using incompressibility,

$$\begin{aligned}&\frac{r_p^2 sm_f}{20}\left(\frac{\partial^2\widetilde{u}_i}{\partial x_j\partial x_j}+\frac{\partial^2\widetilde{u}_j}{\partial x_i\partial x_j}+\frac{\partial^2\widetilde{u}_j}{\partial x_j\partial x_i}\right)\\ &\quad=\frac{r_p^2 sm_f}{20}\left(\frac{\partial^2\widetilde{u}_i}{\partial x_j\partial x_j}+2\frac{\partial}{\partial x_i}\left(\frac{\partial\widetilde{u}_j}{\partial x_j}\right)\right)=\frac{r_p^2 sm_f}{20}\left(\frac{\partial^2\widetilde{u}_i}{\partial x_j\partial x_j}\right)\end{aligned} \tag{1.43}$$

The equation of motion of the sphere can be found through an inverse Laplace transform of equation 1.43. Most terms are straight forward apart from those involving $\sqrt{s}$. As an example we demonstrate the steps to invert $-6\pi r_p^2\sqrt{\mu\rho s}(\widetilde{V}_i-\widetilde{u}_i)$. We seek the inverse Laplace transform given by,

$$\mathscr{L}^{-1}\left[-6\pi r_p^2\sqrt{\mu\rho s}(\widetilde{V}_i-\widetilde{u}_i)\right] \tag{1.44}$$

which can be turned into two solvable problems. The convolution theorem states that the product of two functions in the Laplace domain, $F(s)$ and $G(s)$, is equal to the convolution of the two

functions in the time domain [26]. Or written mathematically, $\mathscr{L}^{-1}[F(s)G(s)] = \mathscr{L}^{-1}[F(s)] \circledast \mathscr{L}^{-1}[G(s)]$. To make use of the convolution theorem we transform,

$$
\begin{aligned}
&-6\pi r_p^2\sqrt{\mu\rho}\left\{\mathscr{L}^{-1}\left[s(\widetilde{V}_i-\widetilde{u}_i)\right]\circledast\mathscr{L}^{-1}\left[\frac{1}{\sqrt{s}}\right]\right\}\\
&\qquad=-6r_p^2\pi\sqrt{\rho\mu}\left(\frac{d(V_i-u_i)}{dt}\circledast\frac{1}{\sqrt{\pi t}}\right)\\
&\qquad=-6r_p^2\sqrt{\rho\mu\pi}\int_{-\infty}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d(V_i-u_i)}{d\tau}d\tau
\end{aligned}
\tag{1.45}
$$

Each term of of the Maxey-Riley equation in both the time and Laplace domains are given in table 1.1. Terms dependent on $\frac{d^2\widetilde{u}_i}{dx_j dx_j}$ have been omitted for clarity, but these can be found in exactly the same way and lead to the Faxén corrections.

term	Laplace	time
added mass	$-\frac{1}{2}(\widetilde{V}_i-\widetilde{u}_i)sm_f$	$-\frac{m_f}{2}\left(\frac{dV_i}{dt}-\frac{du_i}{dt}\right)$
Stokes drag	$-6\pi r_p\mu(\widetilde{V}_i-\widetilde{u}_i)$	$-6\pi r_p\mu(V_i-u_i)$
history term	$-6\pi r_p^2\sqrt{\mu\rho s}(\widetilde{V}_i-\widetilde{u}_i)$	$-6r_p^2\sqrt{\rho\mu\pi}\int_{-\infty}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d(V_i-u_i)}{d\tau}d\tau$

Table 1.1: *The individual force terms of the Maxey-Riley equation are shown in the Laplace and the time domain. Faxén corrections, which are any term containing $\frac{d^2\widetilde{u}_i}{dx_j dx_j}$, can be found similarly.*

By combining the terms in table 1.1 and including the Faxén corrections, we reach the MR equation,

$$
\begin{aligned}
m_p\frac{d\mathbf{V}(t)}{dt} = m_f\frac{D\mathbf{u}}{Dt}+(m_p-m_f)\mathbf{g}-6\pi\mu r_p\left(\mathbf{W}(\mathbf{x},t)-\frac{r_p^2}{6}\Delta\mathbf{u}\right)\\
-\frac{m_f}{2}\left[\frac{d\mathbf{V}}{dt}-\frac{D}{Dt}\left(\mathbf{u}(\mathbf{x},t)+\frac{r_p^2}{10}\Delta\mathbf{u}\right)\right]\\
-6r_p^2\sqrt{\pi\mu}\int_{t_0}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}\left(\mathbf{W}(\mathbf{x},t)-\frac{r_p^2}{6}\Delta\mathbf{u}\right)d\tau,
\end{aligned}
\tag{1.46}
$$

where $\mathbf{W}(\mathbf{x},t)=\mathbf{V}(t)-\mathbf{u}(\mathbf{x},t)$, μ is the dynamic viscosity of the fluid, ρ_f is the density of the fluid and τ represents a historical time in the particles' trajectory. The final term in equation 1.46 is the Basset history force. The Basset history force accounts for the propagation of particle accelerations through the fluid. As such, the term is an integral of the historical accelerations of the particle throughout the particle's history. Similar equations can be found by considering different initial assumptions. These alternative equations extend the validity of Maxey and Riley's work. We will discuss each of the assumptions in turn, along with studies that have derived equations of motion for each case.

Maxey and Riley's first assumptions pertain to the physical properties of the particles themselves. The MR equation is valid for particles that are small, rigid and spherical. Studies exist

that have derived equations of motion for particles of alternative shapes, for example ellipsoidal particles [48]. Others have discussed extensions to particles of finite size [49]. In some cases, for example where one desires to model the distribution of pesticide sprays, the fluid particles are not rigid. For such applications works have aimed to produce equations of motion for deformable particles [50–52]. Of these, Lovalenti and Brady's equation was valid for a deformable and arbitrarily shaped droplet, with finite viscosity [50]. The particles were treated as a fluid with a no-slip boundary between the particle and the external fluid, whilst Michaelides and Feng allowed for interfacial slip between the droplet and the surrounding fluid [52].

Maxey and Riley also assumed that the particle Reynolds number was small, $Re_p \ll 1$. Assuming that the sphere is small makes the derivation simpler, and the resulting equation more usable, but it limits the range of applications. Many attempts have been made to extend the region of validity of the equation beyond the constraint of $Re_p \ll 1$ [53–55] and see [49] for a discussion. A common theme in works that consider finite particle Reynolds numbers is that the form of the Basset force is altered. The Basset force is given by,

$$-6r_p^2\sqrt{\pi\mu}\int_{t_0}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}\left(\mathbf{W}(\mathbf{x},t)-\frac{r_p^2}{6}\Delta\mathbf{u}\right)d\tau, \tag{1.47}$$

and contains the history kernel $\frac{1}{\sqrt{t-\tau}}$. The form of the history kernel is altered when finite Reynolds numbers are considered. Renwei and Mei found a version of the history Kernel for the case of finite Reynolds number [54]. Mei's Kernel decays with t^{-2} rather than the $t^{-1/2}$ of Maxey and Riley's. Mei's kernel is given by,

$$K_{mei}(t)=\left(\left[\frac{\pi t\nu}{r_p^2}\right]^{1/4}+\left[\frac{\pi U^3}{2r_p\nu(0.75+0.211Re_p)^3}t^2\right]^{\frac{1}{2}}\right)^{-2}, \tag{1.48}$$

where, $Re_p=|v(t)-u(\mathbf{x},t)|r_p/\nu$ is the particle Reynolds number, r_p is the radius of the particle, ν is the kinematic viscosity of the fluid and U is the free stream or strong mean velocity [54].

Figure (1.2) shows the Mei kernel cross plotted with the Basset Kernel for three different values of Re_p. It shows that the larger the particle Reynolds number, the faster the decay of the history kernel. In all cases the kernel decays faster than the $1/\sqrt{t}$ Basset Kernel.

Mei's derivation was made for a sphere undergoing periodic oscillations rather than travelling through an arbitrary fluid velocity field. Lovalenti and Brady extended the finite Reynolds number equation to the case of an arbitrary time and space dependent flow [53]; their history kernel decayed at a rate of $t^{-1/2}$ for a step change from a finite to a zero velocity, t^{-2} for a step change from zero to a finite velocity, and exponentially for all cases in between.

In general, extensions to the region of validity of the equation of motion of inertial particles increases the complexity of numerical implementation, and is often unnecessary. Although the formal validity of the MR equation is constrained by the assumption that $Re_p \ll 1$ [47], comparisons between numerical simulations of equation 2.6 and experiments have demonstrated that the MR equation can be acceptable even for finite particle Reynolds numbers. Indeed, after

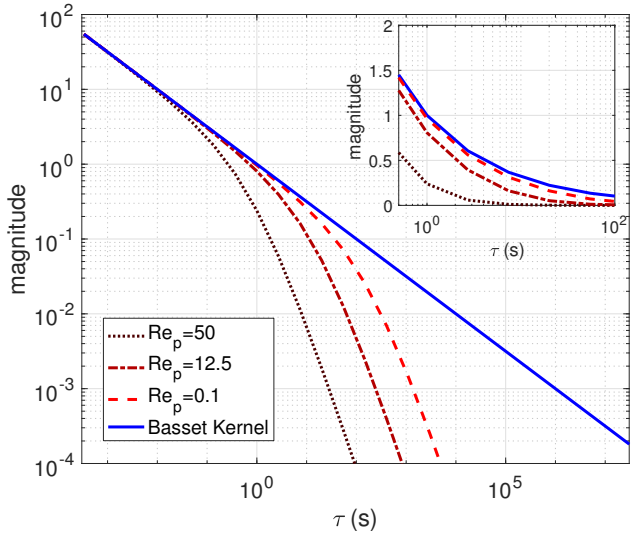


Figure 1.2: *In a comparison between Basset's $1/\sqrt{t}$ kernel, and Mei's kernel calculated for three values of $Re_p > 0$, we can see that the decay rate of the history kernel is influenced by the particle Reynolds number. Mei's Kernel decays faster than the Basset kernel for all finite values of Re_p.*

analysis of the stream function surrounding a sphere at $Re_p = 16.73$, Maxey suggests that equation 1.46 is a reasonable approximation for small finite particle Reynolds numbers [56]. In an experiment in which spherical particles were suspended in an oscillating flow, Esperance [3] demonstrated that the influence of the Basset force could be accurately predicted up to particle Reynolds numbers of 5. In another study, consensus between prediction and experiment was achieved for particle Reynolds numbers of up to 0.5 [57]. Further examples of successful studies demonstrating the accuracy of versions of equation 2.6 derived for finite but small particle Reynolds numbers can be found in [58] and [59]. We take this evidence on board and in our simulations we assume that using equation 2.6 is valid up to a particle Reynolds number of $Re_p = 5$.

In certain contexts the Basset force has a significant affect on the trajectories of inertial particles [3, 57, 60]. Despite its importance, the significant complication associated with the inclusion of the Basset history integral means it is often ignored [4, 61, 62]. Efficient numerical integration schemes for the history force have been developed recently, [30, 63]. Although, as we shall show in the second chapter, are often used in conjunction with numerical methods that are unstable when particles are small.

Given an understanding and method for how a single particle behaves in a fluid, we wish to explore methods for modelling large ensembles of inertial particles. Flows that contain large numbers of particles are a type of two phase flow and are common in nature. Two phase flows which consist of a particle or droplet phase are known as dispersed two phase flows, where the dispersed phase refers to the particles or droplets, and the fluid phase we will refer to as the carrier

phase. Such flows are generally classified based on two quantities. The first is the material density ratio of dispersed phase to the carrier phase, ρ_p/ρ_f. The second quantity is the concentration of dispersed phase, Φ_p. The greater the concentration of particles, the more the dispersed phase influences the carrier phase [22]. For high concentrations of particles ($\Phi_p > 10^{-3}$), collisions between the particles should also be considered [22]. In such cases, modelling interactions between individual particles and capturing the interactions between the carrier and dispersed phases is complicated.

Ardekani and Rangel developed an equation of motion that captures the coupling between two neighbouring spheres [64]. They provided equations which modelled the particles as a system of two interacting particles, although it could be extended to further particles. The disturbance flow caused by a single particle influences the trajectory of the nearby particle leading to coupling terms in the equations of motion. For further particles, coupling terms are required between each possible particle pair. When there are many particles, the equations are hard to use: the number of terms is $O(N^2)$, where N is the number of particles in the system. Although Ardekani's two particle version is apparently successful, it is not suitable for large numbers of spheres.

Often an Eulerian-Lagrangian approach is utilised (see for example [18, 21, 23, 65, 66]). In such cases, the dispersed phase is modelled as a host of individual particles (Lagrangian) and the carrier phase is solved using an Eulerian method such as finite differencing. The influence of the dispersed phase on the fluid is calculated as a balance of forces. The force exerted on each particle is in turn imposed on the fluid with opposite effect. Each discrete cellular volume used for the finite differencing solution of the fluid phase contains a number of particles. The force acting on the fluid phase in each cell is the sum of the reaction forces for each particle in the volume [21, 65]. When the number of individual particles in the dispersed phase is large, the computational cost can become excessive: the computational complexity of pairwise collision detection scales with the number of particles squared; the complexity of particle binning scales as $\mathrm{N_{Cells}} \times \mathrm{N_{particles}}$, where $\mathrm{N_{Cells}}$ is the number of cells in the discretisation and $\mathrm{N_{particles}}$ is the number of particles.

When the concentration of the dispersed phase is large ($\Phi_p > 10^{-3}$), collisions between the particles becomes a concern [22]. For raindrop [15, 19, 67] or planetary formation [68, 69], particle clustering and collisions are fundamental mechanisms by which the bodies grow. In the fourth chapter we investigate methods for modelling collisions between particles. A number of works have investigated collision statistics through direct collision detection, some in homogenous turbulent flows [7, 70], and others in the turbulent wake of a larger particle [69]. In such studies collisions are detected through pairwise collision detection which scales with $O(N^2)$.

Turbulence-particle interaction is a prolific field with an enormous breadth of studies. There is much interest in the way the particles affect the development of the turbulent energy spectrum and a number of different approaches of investigation: from analytical models with homogeneously distributed particles like the two fluid approach [71], to the direct numerical simulations of

Lagrangian particles advected within a numerical fluid simulation [18, 21, 23, 65, 66]. One of the key factors in the influence of the particles on the fluid phase is the preferential concentration of the particles [18].

To summarise, in this thesis we develop novel modelling approaches for inertial particles that are small, rigid and spherical. Although we make the same assumptions as Maxey and Riley, we point out that all our methods can be adapted for equations of motion that are valid for different assumptions. We feel vindicated in this decision because, as discussed, there is empirical evidence that the MR equation is valid for small finite Reynolds numbers. The thesis deals with the following questions which are divided into separate chapters.

Our first port of call is to deal with the Basset history force numerically. In the second chapter we review and point out weaknesses in the contemporary approaches to numerical solutions of the MR equation. We demonstrate that a careful choice of numerical method alleviates issues of instability. Given a robust numerical approach, a parameter space defined by the Stokes number, and density ratio ρ_p/ρ_f, is searched for parameter conditions which require the inclusion of the Basset history force. The parameter space is categorised through comparisons between simulations, experimental data, and analytic solutions of the MR equation.

In the third chapter we develop a method that captures the inertial properties of the particles in an Eulerian sense. The advantage of an Eulerian approach is that there is no computational penalty for increasing the particle concentration. Further, an Eulerian description of inertial particles can be extended to allow feedback between the particle and fluid phase. Our method models particles probabilistically, to give a PDF of particle positions in phase space. We demonstrate that our approach is capable of capturing the inertial properties of the particles in a simple test flow and give a comprehensive review of its success.

In the fourth chapter we show how our transfer operator approach can be extended to consider collisions between the particles. We compare our method with an alternative approach using the eigenvalues of the Cauchy-Green strain tensor, and measure the accuracy of both against the benchmark of a pairwise collision detection scheme. We achieve enormous improvements in efficiency for small losses in accuracy.

In the fifth chapter we investigate how our transfer operator method performs in turbulent flows. Our turbulent flow is a good analogue for the flows in a cloud or a turbulent river, allowing us to give specific focus to two cases: raindrop formation and sedimentation of sand in a river. We show that our approach accurately predicts the sedimentation rates of particles and captures the preferential clustering of the particles for certain parameter regimes. We also investigate the effect of the Basset history force on particle clustering in a decaying turbulent flow and compare our results to those of recent studies by Daitche, Olivieri et al and Guseva et al [7, 31, 72].

CHAPTER 2

NUMERICAL INTEGRATION OF INERTIAL PARTICLE TRAJECTORIES

2.1 Introduction

To better understand the plethora of processes that involve inertial particle advection, the understanding of the advection of the particles themselves must be improved. Regimes in which the density of the particles differs greatly from that of the fluid, such as in cloud microphysics, produce different behaviour from regimes where the densities are more similar, such as in sedimentation.

Particle dynamics vary markedly depending on their length scale, which means models that describe them must be chosen carefully: very small particles are described by stiff equations which can present problems with numerical stability, and larger particles can invalidate the use of the classic differential equation for particle advection entirely. In this chapter we explore the limitations of contemporary numerical descriptions of particle motion and provide recommendations for how the dynamics of particles should be described across a well resolved two dimensional parameter space. The parameter space is defined by particle size, a measure of the degree for which a particle will follow fluid streamlines, and material particle density. This parameter space covers most applications of small spherical particles advected within a fluid.

In many applications, a carefully chosen version of the Basset-Bousinesq-Oseen equation will sufficiently describe the behaviour of inertial particles. Versions have been derived which are appropriate for finite particle Reynolds numbers [53, 54] or which account for particle deformation [50]. Under the assumption that particles are small, rigid and spherical, trajectories can be described by Maxey and Riley's version, given in equation 2.6. In its entirety, the Maxey-Riley equation includes an integral term known as the history force. The history force, or Basset integral, accounts for the delay in propagation of particle accelerations through the fluid. When the particle accelerates relative to the fluid, the fluid immediately adjacent to it responds imposing

a force on the particle. The relative acceleration propagates away from the particle through the fluid surrounding it. Consequently, relative accelerations from the particle's history affect the particle in its present.

Due to the complexity of an adequate numerical implementation, the Basset or history force, is often neglected [63]. When particles are small [4], or density ratios large [7], it is generally considered safe not to include history in modelling. However, there is strong evidence for its inclusion in many contexts. Daitche showed that in a flow around a cylinder, the history force makes a substantial difference to the trajectory of a particle [73]. He also demonstrated that history reduces the degree of particle clustering within a chaotic flow [7], an observation in agreement with findings in an earlier work from Olivieri et al. [31]. Guseva et al, showed that including memory effects can change the number of attractors and their locations in a chaotic flow [72].

For a particle of radius r_p, and density ρ_p travelling in a fluid of density ρ_f, dynamic viscosity μ and velocity $\mathbf{u}(x,t)$, the full MR equation is given by,

(2.1)
$$\rho_p \frac{d\mathbf{v}}{dt} = \rho_f \frac{D\mathbf{u}(x,t)}{Dt} + (\rho_p - \rho_f)\mathbf{g} - \frac{9\mu}{2r_p^2}\hat{\mathbf{w}}(x,t) - \frac{\rho_f}{2}\left[\frac{d\mathbf{v}}{dt} - \frac{D}{Dt}\left(\mathbf{u}(x,t) + \frac{r_p^2}{10}\Delta\mathbf{u}(x,t)\right)\right] - \frac{9\sqrt{\rho_f\mu}}{2r_p\sqrt{\pi}}\left[\int_{t_0}^{t} \frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\hat{\mathbf{w}}(x,\tau))d\tau + \frac{\hat{\mathbf{w}}(x,0)}{\sqrt{t-t_0}}\right],$$

where $\mathbf{g}$ is acceleration due to gravity, t is the current time and τ represents a time in the particle's history. For simplicity we have introduced, $\hat{\mathbf{w}}(x,t) = \mathbf{v}(t) - \mathbf{u}(x,t) + \frac{r_p^2}{6}\Delta\mathbf{u}(x,t)$, where $\mathbf{v}(t)$ is the velocity of the particle. The individual force terms on the right hand side of equation 2.1, from left to right, are as follows: force exerted on the particle by the undisturbed fluid; the buoyancy force; Stokes drag term; the added mass force due to work done by the particle in moving parcels of fluid from its path; and the Basset history force. Terms dependant on $\Delta\mathbf{u}(\mathbf{x},t)$ are known as Faxén corrections. In general the MR equation is valid only when particle radii are small relative to the length scales of fluid motion. As all Faxén terms are proportional to r_p^2 it is often safely neglected [73], although it may be important at the upper limits of validity of the MR equation [47]. In this work we will specifically focus on the influence of the history force, drag force and force from the undisturbed flow. Therefore, we neglect the Faxén terms and replace $\hat{\mathbf{w}}(x,t) = \mathbf{v}(t) - \mathbf{u}(x,t) + \frac{r_p^2}{6}\Delta\mathbf{u}(x,t)$ with the simpler, $\mathbf{w}(x,t) = \mathbf{v}(t) - \mathbf{u}(x,t)$. Dividing both sides of equation 2.1 by the particle density ρ_p gives,

(2.2)
$$\frac{d\mathbf{v}}{dt} = \frac{\rho_f}{\rho_p}\frac{D\mathbf{u}(x,t)}{Dt} + \frac{(\rho_p - \rho_f)}{\rho_p}\mathbf{g} - \frac{9\mu}{2r_p^2\rho_p}\mathbf{w}(x,t) - \frac{\rho_f}{2\rho_p}\left[\frac{d\mathbf{v}}{dt} - \frac{D}{Dt}(\mathbf{u}(x,t))\right] - \frac{9\sqrt{\rho_f\mu}}{2\rho_p r_p\sqrt{\pi}}\left[\int_{t_0}^{t} \frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\mathbf{w}(x,\tau))d\tau + \frac{\mathbf{w}(x,0)}{\sqrt{t-t_0}}\right].$$

Derivatives that appear more than once can be grouped,

$$\left(1+\frac{\rho_f}{2\rho_p}\right)\frac{d\mathbf{v}}{dt}=\left(\frac{\rho_f}{\rho_p}+\frac{\rho_f}{2\rho_p}\right)\frac{D\mathbf{u}(x,t)}{Dt}+\frac{(\rho_p-\rho_f)}{\rho_p}\mathbf{g} \\ -\frac{9\mu}{2r_p^2\rho_p}\mathbf{w}(x,t)-\frac{9\sqrt{\rho_f\mu}}{2\rho_p r_p\sqrt{\pi}}\left[\int_{t_0}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\mathbf{w}(x,\tau))d\tau+\frac{\mathbf{w}(x,0)}{\sqrt{t-t_0}}\right], \tag{2.3}$$

and the coefficient from the left hand side moved to the right,

$$\frac{d\mathbf{v}}{dt}=\frac{\left(\frac{\rho_f}{\rho_p}+\frac{\rho_f}{2\rho_p}\right)}{\left(1+\frac{\rho_f}{2\rho_p}\right)}\frac{D\mathbf{u}(x,t)}{Dt}+\frac{\frac{(\rho_p-\rho_f)}{\rho_p}}{\left(1+\frac{\rho_f}{2\rho_p}\right)}\mathbf{g} \\ -\frac{\frac{9\mu}{2r_p^2\rho_p}}{\left(1+\frac{\rho_f}{2\rho_p}\right)}\mathbf{w}(x,t)-\frac{\frac{9\sqrt{\rho_f\mu}}{2\rho_p r_p\sqrt{\pi}}}{\left(1+\frac{\rho_f}{2\rho_p}\right)}\left[\int_{t_0}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\mathbf{w}(x,\tau))d\tau+\frac{\mathbf{w}(x,0)}{\sqrt{t-t_0}}\right]. \tag{2.4}$$

Some algebraic manipulation is then conducted to simplify the coefficients,

$$\frac{d\mathbf{v}}{dt}=\frac{3(2\rho_f)}{2(2\rho_p+\rho_f)}\frac{D\mathbf{u}(x,t)}{Dt}+\left(1-\frac{3(2\rho_f)}{2(2\rho_p+\rho_f)}\right)\mathbf{g}- \\ \frac{9\mu(2\rho_f)}{2\rho_f r_p^2(2\rho_p+\rho_f)}(\mathbf{w}(x,t))- \\ \frac{9\sqrt{\mu}(2\rho_f)}{2r_p\sqrt{\rho_f\pi}(2\rho_p+\rho_f)}\left[\int_{t_0}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\mathbf{w}(x,\tau))d\tau+\frac{\mathbf{w}(x,0)}{\sqrt{t-t_0}}\right]. \tag{2.5}$$

Equation 2.5 can be dramatically simplified by introducing two key parameters: the density parameter $R=2\rho_f/(2\rho_p+\rho_f)$ and a size parameter $S=\frac{2\rho_f r_p^2}{9\mu}$. Introducing the particle parameters R and S, the MR equation simplifies to,

$$\frac{d\mathbf{v}}{dt}=\frac{3R}{2}\frac{D\mathbf{u}(x,t)}{Dt}+\left(1-\frac{3R}{2}\right)\mathbf{g} \\ -\frac{R}{S}\mathbf{w}(x,t)-\frac{3R}{\sqrt{2\pi S}}\left[\int_{t_0}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\mathbf{w}(x,\tau))d\tau+\frac{\mathbf{w}(x,0)}{\sqrt{t-t_0}}\right]. \tag{2.6}$$

The size parameter has dimensions of time. A dimensionless parameter that is independent of flow characteristics can be found by defining a characteristic length scale L, and a characteristic velocity U. The characteristic length, L, is a typical length scale of features of a fluid velocity field (e.g. the diameter of vortices); the characteristic velocity U is the average fluid velocity. Together, U and L define a characteristic timescale of a flow, $T=L/U$.

Using, T, particle behaviour can be described by a dimensionless parameter independent of the flow characteristics. The appropriate parameter is the Stokes number, St, which is often defined as a ratio of the particle's characteristic stopping distance, τ_p, to a characteristic timescale of the flow τ_f [74]. With a characteristic time scale defined as $\tau_f=T=L/U$ and the characteristic stopping distance of the particle given by $\tau_p=\frac{S}{R}$, the Stokes number is equal to $St=S/(RT)$. Consider a particle with initial velocity v_0, decelerating in a quiescent fluid. We can find the

stopping distance of the particle by setting $\mathbf{u}(x,t) = 0$ and neglecting the influence of history and buoyancy. In a quiescent flow, the MR equation simplifies to,

$$\ddot{x}(t) = -\frac{R}{S}\dot{x}(t). \tag{2.7}$$

The solution of equation 2.7 is,

$$x(t) = \frac{v_0 S}{R}\left(1 - \exp\left(-\frac{Rt}{S}\right)\right) + x_0 \tag{2.8}$$

where v_0 is the initial velocity of the particle and x_0 is its initial position. It is now possible to see that the long time stopping distance of the particle, $t->\infty$, is given by $x(\infty) = v(0)\frac{S}{R}$.

In a fluid that is not quiescent, inertial particles do not follow fluid streamlines. A velocity difference exists between particle and fluid. The dimensionless parameter, St gives an idea of how quickly the velocity difference $\mathbf{v}(t) - \mathbf{u}(x,t)$ decays relative to the timescale of the fluid. Therefore, $St = S/(RT)$ indicates the degree to which an inertial particle will follow fluid streamlines. Particles with a very small Stokes number will behave as passive tracers and follow the flow exactly. Whereas particles with very large Stokes numbers will ignore flow features: they do not respond to fluid accelerations at the rate at which the fluid evolves [74].

With an idea about how the equation of motion links to aspects of particle dynamics, the proceeding chapter introduces a hierarchy of approximations to equation 2.6 that can be exploited to provide computationally tractable solutions. The sensitivity of results to these approximations is shown to be highly context-dependent. Then, an appropriate method for the numerical integration of the MR equation is constructed including a method for numerical approximation of the history force. The stability of the scheme is investigated and then its accuracy is explored using experimental results and analytical solutions to the equation 2.6. Using this appropriate numerical method, an investigation is conducted into the influence of the history force across a well resolved 2 dimensional parameter space. From this, recommendations can be made as to the appropriate use of each of the equations of motion. To concentrate on the influence of the other forces, the buoyancy force will be neglected throughout this chapter.

We consider four members of the hierarchy of approximations to equation 2.6: the passive tracer, the inertial equation, the active tracer, and full particle history in equation 2.6. The passive tracer is described by,

$$\mathbf{v}(t) = \mathbf{u}(\mathbf{x},t), \tag{2.9}$$

where $\mathbf{v}(t)$ is the particle velocity and $\mathbf{u}(\mathbf{x},t)$ is the velocity of the fluid. A passive tracer will follow the undisturbed fluid streamlines exactly, and is approximately valid in cases of neutral buoyancy where the particle and fluid densities are equal, and when particle diameters are sufficiently small that their relative inertia to the fluid can be ignored.

Where inertia plays a role, the simplest equation able to qualitatively capture some important aspects of behaviour is given by,

$$\dot{\mathbf{v}}(t) = \frac{3}{2}R\frac{D\mathbf{u}}{Dt} - \frac{R}{S}(\mathbf{v}(t) - \mathbf{u}(\mathbf{x},t)). \tag{2.10}$$

Although this formulation excludes the integral term due to history, it still accounts for inertial influences arising from both the fluid-particle density ratio, R, and particle diameter given through the size parameter S. When particle diameters are small, and density ratios large, the drag term dominates and the influence of the history force is small by comparison. In this parameter regime, the active tracer will be shown to work well.

The relation between the active and passive tracer can be investigated in detail through the introduction of the 'inertial equation'. It turns out that for small $\frac{S}{R}$ values the active tracer is just a correction to the passive tracer that is attracted to a three-dimensional manifold. This relation has been shown in [4], which we recall here. Equation (2.10) can be rescaled by setting $t = \tau\frac{R}{S}$ under the assumption that $\frac{S}{R}$ is small. Introducing τ allows us to separate the dynamics of the system into those that change over short time periods evolving with τ, and those that change little over changes in τ. If we also include the position $\boldsymbol{x}$ of the particle and define $\epsilon = \frac{S}{R}$ then the scaled equation becomes

$$\left.\begin{aligned} \boldsymbol{x}'(\tau) &= \epsilon \boldsymbol{v}(\tau) \\ \boldsymbol{v}'(\tau) &= \epsilon\frac{3R}{2}\frac{D\boldsymbol{u}(\boldsymbol{x},\phi)}{D\phi} - \big(\boldsymbol{v}(\tau) - \boldsymbol{u}(\mathbf{x}(t),\phi(t))\big) \\ \phi'(\tau) &= \epsilon \end{aligned}\right\}, \tag{2.11}$$

where $'$ represents differentiation with respect to τ. When $\epsilon \to 0$, equation (2.11) reduces to

$$\left.\begin{aligned} \mathbf{x}'(\tau) &= \mathbf{0} \\ \mathbf{v}'(\tau) &= -\big(\mathbf{v}(\tau) - \mathbf{u}(\mathbf{x}(t),\phi)\big) \\ \phi'(\tau) &= 0 \end{aligned}\right\},$$

which is called the layer equation and has a manifold of equilibria, called the critical manifold, given by

$$\mathscr{M}_0 = \left\{(\mathbf{x},\mathbf{v},\phi) \in \mathbb{R}^7 : \mathbf{v} = \mathbf{u}(\mathbf{x},\phi)\right\}.$$

The critical manifold $\mathscr{M}_0$ is attracting, which also implies normal hyperbolicity. Due to normal hyperbolicity the critical manifold persists for $0 < \epsilon \ll 1$ sufficiently small. This means that there is a function $\mathbf{h}(\mathbf{x},\phi,\epsilon)$ such that

$$\mathscr{M}_\epsilon = \left\{(\mathbf{x},\mathbf{v},\phi) \in \mathbb{R}^7 : \mathbf{v} = \mathbf{h}(\mathbf{x},\phi,\epsilon)\right\}$$

is an invariant manifold of (2.11) for $0 < \epsilon \ll 1$. The usual way to approximate the function $\boldsymbol{h}$ is by a power series in ϵ such that

$$\mathbf{h}(\mathbf{x},\phi,\epsilon) = \sum_{k=0}^{\infty} \epsilon^k \mathbf{h}_k(\mathbf{x},\phi).$$

The formal series expansion has been carried out in [4] with the following result

$$
\begin{aligned}
\mathbf{h}_0 &= \mathbf{u} \\
\mathbf{h}_1 &= \left(\frac{3R}{2} - 1\right)\frac{D\mathbf{u}}{Dt} \\
\mathbf{h}_k &= -\left[\frac{D\mathbf{h}_{k-1}}{Dt} + \nabla\mathbf{u}\cdot\mathbf{h}_{k-1} + \sum_{i=1}^{k-2}\nabla\mathbf{h}_i\cdot\mathbf{h}_{k-i-1}\right],\ k \geq 2
\end{aligned}
\tag{2.12}
$$

The final equation that tracks the particle is therefore

$$
\dot{\mathbf{x}} = \mathbf{h}(\mathbf{x}, t, \epsilon). \tag{2.13}
$$

One advantage of equation 2.12 is that it has a single time-scale. Therefore, solving it backwards in time eliminates the exponential divergence from the invariant manifold $\mathscr{M}_\epsilon$ present in equation 2.10. This is an advantage when we would like to find Lagrangian coherent structures by calculating finite time Lyapunov exponents (FTLE) (see chapter 4). More specifically, if one aimed to recover the release site of a pollutant into the ocean, then the trajectory of the pollutant would have to be calculated backwards in time. The exponential attraction to a trajectory on the manifold forwards in time becomes exponential divergence backwards in time.

We also note that for neutrally buoyant particles when $R = 2/3$, the manifold simply reduces to $\boldsymbol{h} = \boldsymbol{u}$. This manifold exists for all values of ϵ, but it is no longer attracting for sufficiently large ϵ. Conditions for the manifold remaining globally attractive were determined by Haller in [4].

We note that the history integral $\int_{t_0}^{t} \frac{1}{\sqrt{t-\tau}} \frac{\mathrm{d}}{\mathrm{d}\tau}(\mathbf{v}-\mathbf{u})\mathrm{d}\tau$ cannot be analysed in the same fashion as the inertial term. The history term is not necessarily bounded and if there is an attracting invariant manifold it is certainly not normally hyperbolic [32] and does not persist in the same way as M_0 persists for the active tracer.

The inertial equation will prove useful in chapter 3 where it will be used to significantly improve the computational complexity of one of our methods. In this chapter, we will simply explore its accuracy relative to the active tracer in equation 2.10. We now turn to the numerical solution of the Maxey-Riley equation (2.6).

2.2 Numerical Implementations

2.2.1 Basset force

In this section, we present a method for the numerical integration of the history force, F_B. The method is a modest adaptation of the 2nd order approach used by Hinsberg [63]. The Basset integral, that we seek to approximate, is given by,

$$
F_B = \frac{3R}{\sqrt{2\pi S}} \int_{t_0}^{t} \frac{1}{\sqrt{t-\tau}} g(\tau)\mathrm{d}\tau, \tag{2.14}
$$

where $g(\tau) = \frac{d}{d\tau}(\mathbf{v}(\tau) - \mathbf{u}(\mathbf{x}, \tau)$ and t_0 is the initial time.

For discretisation of equation 2.14 we introduce a series of discrete time parameters. First is the history window, $t_{win} = t - t_0$, which denotes the time over which the Basset integral is defined. Discrete time steps in the particles history are given by $\tau_n = t - n\Delta t$, where Δt is the time-step and $n = 0, 1, 2, 3 \ldots N$ enumerates the time steps from the current time step $n = 0$, to the time step at the end of the window, $n = N$.

Hinsberg's approach for numerically integrating the Basset force is to divide the integral into a sum of integrals defined over a single time step. Within each time step, $g(\tau)$ is replaced by its linear interpolant such that, $g(\tau) \approx g(\tau_n) + (g(\tau_{n-1}) - g(\tau_n))(\tau - \tau_n)/\Delta t$. The notation $g(\tau_n)$ denotes the value of $g(\tau)$ at discrete intervals in time. Following Hinsberg's approach produces,

$$F_B = \frac{3R}{\sqrt{2\pi S}} \sum_{n=1}^{N} \int_{\tau_n}^{\tau_{n-1}} \frac{g(\tau_n) + (g(\tau_{n-1}) - g(\tau_n))(\tau - \tau_n)/\Delta t}{\sqrt{t - \tau}} \mathrm{d}\tau, \tag{2.15}$$

providing a sum of integrals that can be calculated exactly. After carrying out these integrals, the Basset force can be approximated numerically as a summation given by,

$$F_B = \frac{3R\sqrt{\Delta t}}{\sqrt{2\pi S}} \left(\frac{4}{3} g_0 - c_N g_N + \sum_{n=1}^{N-1} g_n c_n \right),$$

where,

$$c_n = \left(\frac{n + \frac{4}{3}}{(n+1)\sqrt{n+1} + (n + \frac{3}{2})\sqrt{n}} + \frac{n - \frac{4}{3}}{(n-1)\sqrt{n-1} + (n - \frac{3}{2})\sqrt{n}} \right) \tag{2.16}$$

$$c_N = \left(\frac{(N - \frac{4}{3})}{(N-1)\sqrt{N-1} + (N - \frac{3}{2})\sqrt{N}} \right).$$

For efficiency, the calculation of the coefficients c_n is conducted once at the start of the simulation. Each time step, the most costly computation is the evaluation of the inner product $\sum_{n=1}^{N-1} g(\tau_n) c_n = \langle g(\tau_n), c_n \rangle$. As the simulation time increases, so does the length of the window. Consequently, the computational cost of the evaluation of equation 2.16 increases with simulation time. In general, for large numbers of particles, or long simulation times, storing and evaluating the entire history of a particle is intractable.

One approach that retains numerical tractability is to constrain the length of historical time used to update the current state of the system. The integrand in the Basset force includes the polynomially decaying history kernel $1/\sqrt{t - \tau}$. Although it might be argued that the effect of the furthest reaches of the particle's history has negligible influence on the present, we find that we cannot ignore the end of the history tail completely. Instead we make a compromise. Following [63], we can account for the recent historical influence by direct computation and approximate the furthest reaches with a fitted curve with a suitably-behaved integral. In the present work we use a variation of Hinsberg's method [63].

The Basset term is split into two integrals: the history window F_{win}, and the exponential

approximation to the tail F_{exp}, to give,

$$F_B = \underbrace{\frac{3R}{\sqrt{2\pi S}}\int_{t-t_{win}}^{t}\frac{g(\tau)}{\sqrt{t-\tau}}\mathrm{d}\tau}_{F_{win}} + \underbrace{\frac{3R}{\sqrt{2\pi S}}\int_{t_0}^{t-t_{win}} K_{exp}(t-\tau)g(\tau),\mathrm{d}\tau}_{F_{exp}} \tag{2.17}$$

where it is assumed that an exponential kernel, $K_{exp}(t)$, can be found that suitably approximates the 'tail' of the polynomial kernel, $1/\sqrt{t-\tau}$. Hinsberg et al. provide one such suitable approximation. In Hinsberg's work, $K_{exp}(t)$ is found in terms of a summation of m exponential kernels. The set of m exponential kernels takes the form,

$$K_{exp}(t) = \sum_{i=1}^{m} a_i K_i(t) \quad \text{with,} \quad K_i(t) = \sqrt{\frac{\mathrm{e}^1}{t_i}}\exp\left(-\frac{t}{2t_i}\right), \tag{2.18}$$

where the fitting parameters, a_i and t_i, must be found. These are optimally determined by Hinsberg for summations up to and including 10 exponentials. It is only necessary for the exponential kernel to accurately approximate the tail of the history kernel. As such, the parameters are optimally defined such that they minimise the difference between the original kernel, $K_B(t) = 1/\sqrt{t}$ and the exponential kernel K_{exp}, for $t-\tau > t_{win}$.

First, Hinsberg finds the Euclidian distance between the Basset force, F_B, and the approximation made using the exponential kernel, F_{exp},

$$\left|\frac{F_B - F_{exp}}{\frac{3R}{\sqrt{2\pi S}}}\right| = \left|\int_{-\infty}^{t-t_{win}} K_B(t-\tau)g(\tau) - \int_{-\infty}^{t-t_{win}} K_{exp}(t-\tau)g(\tau)d\tau\right| = \left|\int_{-\infty}^{t-t_{win}} (K_B - K_{exp})(t-\tau)g(\tau)d\tau\right|. \tag{2.19}$$

which through integration by parts yields,

$$\left|\frac{F_B - F_{exp}}{\frac{3R}{\sqrt{2\pi S}}}\right| = \left(\left|K_B(t_{win}) - \widetilde{K}_{exp}(t_{win})f(t-t_{win})\right| + \int_{t_{win}}^{\infty}\left|\frac{d(K_B - \widetilde{K}_{exp})(t)}{dt}f(t)\right|dt\right). \tag{2.20}$$

Consider that $f(t)$ is bounded for all time by $f(t) \leq ||\mathbf{v}-\mathbf{u}||_\infty$, where $||\cdot||_\infty$ is the infinity norm. An upper bound for equation 2.22 can be found by factoring out the upper bound for $f(t)$ from both terms,

$$\left|\frac{F_B - F_{exp}}{\frac{3R}{\sqrt{2\pi S}}}\right| \leq ||\mathbf{v}-\mathbf{u}||_\infty\left(\left|K_B(t_{win}) - \widetilde{K}_{exp}(t_{win})\right| + \int_{t_{win}}^{\infty}\left|\frac{d(K_B - \widetilde{K}_{exp})(t)}{.}dt\right|dt\right) \tag{2.21}$$

Time is scaled by t_{win} to make K_{exp} independent of the window length, and the magnitudes of the kernels are scaled by $K_B(t_{win})$ to give: $\widetilde{t} = t/t_{win}$, $\widetilde{t}_i = t_i/t_{win}$ and $\widetilde{K}_B = \frac{K_B(t)}{K_B(t_{win})}$. Because $||\mathbf{v}-\mathbf{u}||_\infty$ is a function that depends on the flow and parameters of the particles, it does not influence the calculation of $K_{exp}(t)$, and we are left to minimise,

$$\left\{\left|1 - \widetilde{K}_{exp}(1)\right| + \int_1^{\infty}\left|\frac{d(\widetilde{K}_B - \widetilde{K}_{exp})(\widetilde{t})}{dt}\right|dt\right\}, \tag{2.22}$$

where $K_B(t_{win})/K_B(t_{win})$ is equal to 1. We now seek a set of exponential kernels of the form,

$$\widetilde{K}_{exp}(t) = \sum_{i=1}^{m} a_i \widetilde{K}_i(\widetilde{t}) \quad \text{with,} \quad \widetilde{K}_i(\widetilde{t}) = \sqrt{\frac{e}{\widetilde{t}_i}} exp\left(-\frac{\widetilde{t}}{2\widetilde{t}_i}\right). \tag{2.23}$$

that minimise equation 2.22.

The minimisation method begins by choosing values of $\widetilde{t}_i$ that are suitably spaced to match the rate of change of the history kernel. For low values of $\widetilde{t}_i$ the kernel has a steep gradient and so values must be close together to get a good approximation. For larger values the gradient is shallower, and fewer points are needed to approximate K_B. Given a reasonable range of values for t_i the corresponding values of a_i are found by minimising equation 2.22. First, the integral in equation 2.22 is evaluated, and using the chosen values of t_i, a minimum of the function can be found in terms of the values for a_i. Optimum values of a_i and $\widetilde{t}_i$ calculated by Hinsberg are presented in Table (2.1). We have verified these values as a minimum of equation 2.22.

i	$\widetilde{t}_i$	a_i
1	0.23477481312586	0.1
2	0.28549576238194	0.3
3	0.28479416718255	1
4	0.26149775537574	3
5	0.32056200511938	10
6	0.35354490689146	40
7	0.39635904496921	190
8	0.42253908596514	1000
9	0.48317384225265	6500
10	0.63661146557001	50000

Table 2.1: *Hinsberg found values for a_i and $\widetilde{t}_i$ which minimise equation 2.22. His values are presented here for* 10 *exponentials, $m = 10$, which we have verified.*

Armed with values for a_i and t_i, a method for numerical integration of the exponential approximation must be found. Again Hinsberg provides an approach. The tail is approximated by taking the linear interpolant of the tail for the time period ($t - t_{win} - \Delta t$ to $t - t_{win}$) (the time step immediately after the end of the window). The remaining contribution of the tail can be included by recursively adding the tail section. See [63] for details. It turns out that the second integral can be evaluated using a method that is more tractable to higher order methods than Hinsberg's. We present our alternative approach in the following lines.

By defining the exponential tail integral as κ_i

$$\kappa_i = \int_{t_0}^{t-t_{win}} a_i \sqrt{\frac{\mathrm{e}}{t_i}} exp\left(-\frac{(t-\tau)}{2t_i}\right) g(\tau) d\tau, \tag{2.24}$$

we can introduce a differential equation for $\dot{\kappa}_i$,

$$\dot{\kappa}_i = \int_{t_0}^{t-t_{win}} K_{exp}(t-\tau) g'(\tau) d\tau \tag{2.25}$$

where we have used the fact that the derivative of a convolution is, $d/dt(\int_{-\infty}^{\infty} F(t-\tau)G(\tau)d\tau) = \int F(t-\tau)G'(\tau)d\tau$ for arbitrary functions $F(t)$ and $G(t)$. We expand $\dot{\kappa}_i$ through integration by parts to give,

$$\dot{\kappa}_i = a_i\sqrt{\frac{e}{t_i}}\exp\left(-\frac{(t-(t-t_{win}))}{2t_i}\right)g(t-t_{win}) - \frac{1}{2t_i}\int_{t_0}^{t-t_{win}} a_i\sqrt{\frac{e}{t_i}}exp\left(-\frac{(t-\tau)}{2t_i}\right)g(\tau)d\tau. \tag{2.26}$$

We take advantage of the fact that $\dot{\kappa}$ is now a function of κ,

$$\dot{\kappa}_i = a_i\sqrt{\frac{e}{t_i}}exp\left(-\frac{t_{win}}{2t_i}\right)g(t-t_{win}) - \frac{1}{2t_i}\kappa_i. \tag{2.27}$$

equation 2.27 can now be solved numerically alongside the the rest of the MR equation to give κ. Finally, the full system of equations that we aim to solve numerically can be presented. We write the MR equation as a system of ordinary differential equations together with our approximations for the Basset force,

(2.28)

$$\begin{aligned}
\dot{\mathbf{x}}(t) =& \mathbf{v}(t)\\
\dot{\mathbf{v}}(t) =& \frac{3R}{2}\frac{D\mathbf{u}}{Dt} - \frac{R}{St}(\mathbf{v}(t)-\mathbf{u}(\mathbf{x},t)) - \frac{3R}{\sqrt{2\pi St}}\left(\sum_{i=1}^{m}\kappa_i + \frac{4}{3}\sqrt{\Delta t}(\dot{\mathbf{v}}(t)-\dot{\mathbf{u}}(\mathbf{x},t)) + \sqrt{\Delta t}\sum_{n=1}^{N} c_n g(\tau_n)\right)\\
\dot{\kappa}_i =& \beta_i g(\tau_{N+1}) - \frac{\kappa_i}{2t_i},
\end{aligned}$$

where, m is the number of exponentials used to approximate the tail, and β_i is given by $\beta_i = a_i\sqrt{\frac{e}{t_i}}exp\left(-\frac{t_{win}}{2t_i}\right)$.

2.2.2 Numerical methods

The choice of numerical method to integrate equation 2.28 requires further consideration. In previous works on the subject the numerical methods chosen to integrate equation 2.6 were the second and third order Adams-Bashforth linear multistep methods [30, 63]. We will show that the stability of the second order method used by Hinsberg is dependent on the step size Δt. We will then introduce an alternative which does not have the same issues of stability, namely the 2nd order implicit backwards differentiation formula (BDF2) in equation 2.32. We choose the BDF2 because it was specifically designed for solving stiff systems of equations and the MR equation is an example of such a system [75].

For a system of equations of the form $\dot{\mathbf{x}} = \mathbf{v}$, $\ddot{\mathbf{x}} = f(t,\mathbf{x},\mathbf{v})$ the 2nd order Adams-Bashforth method is given by,

$$\begin{aligned}
\mathbf{x}_{n+1} &= \mathbf{x}_n + \frac{\Delta t}{2}(3\mathbf{v}_n - \mathbf{v}_{n-1}))\\
\mathbf{v}_{n+1} &= \mathbf{v}_n + \frac{\Delta t}{2}(3f(t_n,\mathbf{x}_n,\mathbf{v}_n) - f(t_{n-1},\mathbf{x}_{n-1},\mathbf{v}_{n-1}))\\
t_{n+1} &= t + \Delta t,
\end{aligned} \tag{2.29}$$

where $\mathbf{x}_n$ is the current state of the system, Δt is the time-step and t_n is the current discrete time. The subscript $_n$ refers to the nth time step. We use the same time-step, Δt for the full scheme, as for integration of the Basset integral. Although different time-steps could be used for each, we use the same time step for both to ensure the same level of accuracy for the integration of the Basset force and the particle tracking.

To minimise notational clutter, we allow f_n to represent $f(t_n, \mathbf{x}_n, \mathbf{v}_n)$; similarly, $\mathbf{u}_n$ implies $\mathbf{u}(t_n, \mathbf{x}_n)$. For a linear system such as the active tracer given by equation 2.10, the stability conditions of the Adams-Bashforth method can be determined. We now substitute the active tracer in equation 2.10 for the function f_n to give the matrix equation,

$$\begin{bmatrix} \mathbf{x}_{n+1} \\ \mathbf{v}_{n+1} \\ \mathbf{x}_n \\ \mathbf{v}_n \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \frac{3\Delta t}{2} & 0 & \frac{\Delta t}{2} \\ 0 & 1 - \frac{3R\Delta t}{2S} & 0 & \frac{R\Delta t}{2S} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{update matrix}} \begin{bmatrix} \mathbf{x}_n \\ \mathbf{v}_n \\ \mathbf{x}_{n-1} \\ \mathbf{v}_{n-1} \end{bmatrix} + \frac{\Delta t}{2} \underbrace{\begin{bmatrix} 0 \\ \frac{3R}{2}\left(3\frac{Du_n}{Dt} + \frac{Du_{n-1}}{Dt}\right) + \frac{R}{S}(3u_n - u_{n-1}) \\ 0 \\ 0 \end{bmatrix}}_{\text{bounded functions}} \tag{2.30}$$

The functions of $\mathbf{u}_n$ can be assumed to be bounded in a realistic fluid flow for all $\mathbf{x}_n$, and therefore will not affect the stability of the method. For stability of the method, we require only that the eigenvalues of the update matrix lie within the unit circle in the complex plane. The eigenvalues of the update matrix are given by,

$$\Lambda = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 - (3R\Delta t)/(2S) \end{bmatrix}, \tag{2.31}$$

which all lie within the unit circle provided that the following condition is satisfied. **Condition 1:** For stability on the global error we require that $|1 - 3\Delta t\frac{R}{S}/2| \leq 1$, or more simply, $0 \leq \Delta t\frac{R}{S} \leq 4/3$.

From **Condition 1**, we can see that Δt must be reduced to maintain numerical stability as particle diameters get smaller, or the densities of the particle and fluid become increasingly similar. If one desires to simulate an ensemble of particles with differing parameter values, or perhaps model a particle growth process, one does not want to have to reduce the step size of the solver to maintain stability. In the following we demonstrate that using an implicit method solves this problem.

We now test the second order implicit BDF2, which, when constructed to solve a system of equations of the form $\dot{\mathbf{x}} = \mathbf{v}$, $\ddot{\mathbf{x}} = f(t, \mathbf{x}, \mathbf{v})$, is given by [76],

$$\begin{aligned} \mathbf{x}_{n+2} &= \frac{4}{3}\mathbf{x}_{n+1} - \frac{1}{3}\mathbf{x}_n + \frac{2\Delta t}{3}\mathbf{v}_{n+2} \\ \mathbf{v}_{n+2} &= \frac{4}{3}\mathbf{v}_{n+1} - \frac{1}{3}\mathbf{v}_n + \frac{2\Delta t}{3} f(t_{n+2}, \mathbf{x}_{n+2}, \mathbf{v}_{n+2}), \\ t_{n+2} &= t_{n+1} + \Delta t \end{aligned} \tag{2.32}$$

and after substituting in the active tracer in equation 2.10, can be written in the matrix form,

$$\underbrace{\begin{bmatrix} 1-2\frac{\Delta t}{3} & 0 & 0 & 0 \\ 0 & 1+\frac{2R\Delta t}{3St} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\Upsilon} \begin{bmatrix} \mathbf{x}_{n+2} \\ \mathbf{v}_{n+2} \\ \mathbf{x}_{n+1} \\ \mathbf{v}_{n+1} \end{bmatrix} \tag{2.33}$$
$$= \underbrace{\begin{bmatrix} \frac{4}{3} & 0 & -\frac{1}{3} & 0 \\ 0 & \frac{4}{3} & 0 & -\frac{1}{3} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}}_{\Psi} \begin{bmatrix} \mathbf{x}_{n+1} \\ \mathbf{v}_{n+1} \\ \mathbf{x}_{n} \\ \mathbf{v}_{n} \end{bmatrix} + \frac{2\Delta t}{3} \underbrace{\begin{bmatrix} 0 \\ \frac{3R}{2}\frac{Du_{n+2}}{Dt} + \frac{R}{S}u_{n+2} = \lambda 1 \\ 0 \\ 0 \end{bmatrix}}_{\text{bounded functions}}$$

where all linear implicit terms have been moved to the left hand side of the equation. By implicit term, we mean any term dependant on x_{n+2} or v_{n+2}. Next we multiply both sides by the inverse of Υ to find an update matrix,

$$\begin{bmatrix} \mathbf{x}_{n+2} \\ \mathbf{v}_{n+2} \\ \mathbf{x}_{n+1} \\ \mathbf{v}_{n+1} \end{bmatrix} = \underbrace{\Upsilon^{-1}\Psi}_{\text{update matrix}} \begin{bmatrix} \mathbf{x}_{n+1} \\ \mathbf{v}_{n+1} \\ \mathbf{x}_{n} \\ \mathbf{v}_{n} \end{bmatrix} + \frac{2\Delta t}{3}\Upsilon^{-1} \underbrace{\begin{bmatrix} 0 \\ \frac{3R}{2}\frac{Du_{n+2}}{Dt} + \frac{R}{S}u_{n+2} \\ 0 \\ 0 \end{bmatrix}}_{\text{bounded functions}}. \tag{2.34}$$

As before, for numerical stability, we require all eigenvalues of the update matrix to lie in the unit circle in the complex plane. The eigenvalues of $\Upsilon^{-1}\Psi$ are given in equation 2.35.

$$\Theta = \begin{bmatrix} 1/3 \\ 1 \\ \frac{2S-\sqrt{S^2-2R\Delta tS}}{3S+2R\Delta t} \\ \frac{2S+\sqrt{S^2-2R\Delta tS}}{3S+2R\Delta t} \end{bmatrix} \tag{2.35}$$

To ensure we identify all relevant solutions of our stability condition, $\max|\Theta_i| \leq 1$, $i \in [1,4]$, we conduct a numerical sweep through the space of all reasonable, positive values of S, R, and time step Δt: $0 \leq St \leq 10$; $0 \leq R \leq 2$; $0.001 \leq \Delta t \leq 1$. For this parameter range, the eigenvalues of the update matrices for the BDF2, and the 2nd oreder Adams Bashforth method, are plotted in figure 2.1. It is clear that for this range of parameter values, the BDF2 is unconditionally stable. Clearly the BDF2 is more suited to solution of the MR equation than the second order Adams Bashforth method.

Given knowledge that the BDF2 is a suitable numerical method for solution of the MR equation, we reintroduce our numerical approaches for dealing with the history force F_B. Turning back to equation 2.28, we can see that $\dot{\mathbf{v}}(t)$ is a function of itself. Through simple rearrangement

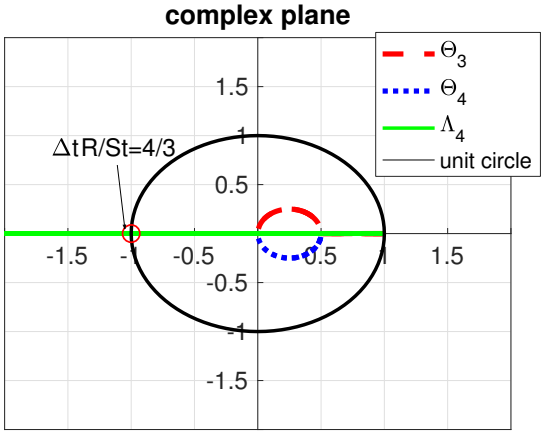


Figure 2.1: *The eigenvalues of the update matrices from the BDF2 and Adams-Bashforth method indicate the stability of the schemes. Because stability is a function of S, R and Δt, a large range of values are plotted for each: $0 \leq St \leq 10$; $0 \leq R \leq 2$; $0.001 \leq \Delta t \leq 1$. Eigenvalues for the BDF2 are given by Θ_i, which lie in the unit circle unconditional on values in our parameter range. Values of Λ, which are the eigenvalues for the Adams-Bashforth method, stray outside the unit circle when $\Delta tR/S > 4/3$. At this point the Adams-Bashforth method becomes unstable.*

and the introduction of $\zeta = 4R\sqrt{\Delta t/(2\pi S)}$, this implicit term is moved to the L.H.S leading to,

$$(1+\zeta)\dot{\mathbf{v}}_n = \frac{3R}{2}\frac{D\mathbf{u}_n}{Dt} + \frac{R}{S}(\mathbf{u}_n - \mathbf{v}_n) - \frac{3R}{\sqrt{2\pi S}}\left(\sum_{i=1}^{m}\kappa_i - \frac{4}{3}\sqrt{\Delta t}\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{v}_n.\nabla\mathbf{u_n}\right) + \sqrt{\Delta t}\sum_{i=1}^{N} c_i g_i\right). \tag{2.36}$$

where we have also expanded $\frac{d\mathbf{u}}{dt}$ into $\frac{\partial \mathbf{u}}{\partial t} + \mathbf{v}.\nabla\mathbf{u}$ to elucidate all implicit components of the equation of motion. A second order implicit method such as the BDF2 requires $f(t_{n+2}, \mathbf{x}_{n+2}, \mathbf{v}_{n+2}) = f_{n+2} = [\mathbf{v}_{n+2}, \dot{\mathbf{v}}_{n+2}]$ which for the full MR equation is given by,

$$f_{n+2} = \begin{bmatrix} \mathbf{v}_{n+2} \\ \frac{1}{1+\zeta}\left(\frac{3R}{2}\frac{D\mathbf{u}_{n+2}}{Dt} + \frac{R}{S}(\mathbf{u}_{n+2} - \mathbf{v}_{n+2}) - F_{exp} - \zeta\left(\frac{\partial \mathbf{u}_{n+2}}{\partial t} + \mathbf{v}_{n+2}.\nabla\mathbf{u}_{n+2} + \frac{3}{4}\sum_{i=1}^{N} c_i g_i\right)\right) \end{bmatrix} \tag{2.37}$$

Substituting f_{n+2} into equation 2.32 gives,

$$\begin{bmatrix} \mathbf{I}, & -\frac{2\Delta t}{3}\mathbf{I} \\ \mathbf{0}, & \frac{2\Delta t}{3(1+\zeta)}\left(\frac{R}{St} - \zeta\nabla\mathbf{u}_{n+2}\right) \end{bmatrix} \begin{bmatrix} \mathbf{v}_{n+2} \\ \mathbf{x}_{n+2} \end{bmatrix} = \frac{4}{3}\begin{bmatrix} \mathbf{v}_{n+1} \\ \mathbf{x}_{n+1} \end{bmatrix} - \frac{1}{3}\begin{bmatrix} \mathbf{v}_n \\ \mathbf{x}_n \end{bmatrix} + \frac{2\Delta t}{3} f_{RHS}, \tag{2.38}$$

where all terms that are linearly dependent on $\mathbf{v}_{n+2}$ have been moved to the left hand side and,

$$f_{RHS} = \begin{bmatrix} 0 \\ \frac{1}{1+\zeta}\left(\frac{3R}{2}\frac{D\mathbf{u}_{n+2}}{Dt} + \frac{R}{S}\mathbf{u}_{n+2} - f_{exp} - \zeta\frac{\partial \mathbf{u}_{n+2}}{\partial t} - \frac{3\zeta}{4}\sum_{i=1}^{N} c_i g_i\right) \end{bmatrix} \tag{2.39}$$

Our fluid velocity field $\mathbf{u}_{n+2}$ could be any nonlinear function of $\mathbf{x}_{n+2}$ and t provided it defines an incompressible and continuous velocity field. We cannot yet calculate $\mathbf{u}_{n+2}$, because we have

not yet calculated the particle position $\mathbf{x}_{n+2}$. The solution is to find $\mathbf{x}_{n+2}$ by integrating the velocity with an explicit step and then feeding the new position back into equation 2.38 in a predictor-corrector type method. The stability problems with the explicit method we discussed previously do not arise when integrating the velocity to find position. We use the third order explicit Adams-Bashforth scheme [77] to find $\mathbf{x}_{n+2}$ given by,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \left[\frac{23}{12}\mathbf{v}_n - \frac{4}{3}\mathbf{v}_{n-1} + \frac{5}{12}\mathbf{v}_{n-2} \right]. \tag{2.40}$$

2.3 Method Verification

In order to test the validity of the scheme in equation 2.38, a comparison is made with an analytical solution of the Maxey-Riley Equation. For the case of rigid body rotation an analytical solution is found by Candelier to compare with his experiment [60]. The same solution was used by Daitche to assess the validity of higher order numerical schemes [30]. It is used again here to validate our numerical scheme.

We first define the flow $U_r = \mathbf{x} \times \omega$, where $\mathbf{x}$ is the position of a particle, ω is the angular velocity of the flow and $\times$ denotes the cross product. An aerosol particle suspended within the flow U_r, will migrate away from the centre of the vortex. An exact expression for the trajectory of an aerosol is found by Candelier in the form of the complex variable $Z(t) = \Re(Z(t)) + \Im(Z(t))$, where $\Re(Z)$ represents the radial migration of the particle, and $\Im(Z(t))$, the angular position. The analytic solution is given by,

$$Z(t) \approx 2\frac{A_1}{X_1} e^{X_1^2 t} + Z(0), \tag{2.41}$$

where A_1 and X_1 are constants dependent on the initial position and velocity of the particle, and on the values of S and R. See [60] for a full derivation and explanation. Using this solution an estimate for the error of the numerical method can be obtained.

Figure (2.2a) shows a comparison between the trajectories calculated with equation 2.41, and numerical trajectories of an aerosol particle advected by U_r. The time step Δt was changed between $0.1, 0.01$, and 0.001 to demonstrate the qualitative difference in accuracy with step size. The numerical trajectories were simulated using the scheme in equation 2.38.

Figure (2.2b) shows how the global error $\epsilon = max_{(t_{max}=100)}|Z(t) - x(t, \Delta t)|$, and error per time step, $(\Delta t/t_{max})\epsilon$, scale with Δt. The method is second order with respect to the analytical solution given by Candelier.

A further consideration is that of numerical efficiency. A key motivation for using an exponential approximation to the tail of the history force is to reduce the number of computations required to simulate trajectories. Figure 2.3a demonstrates how using the exponential tail increases computational efficiency. A history integration over the entire particle history grows with

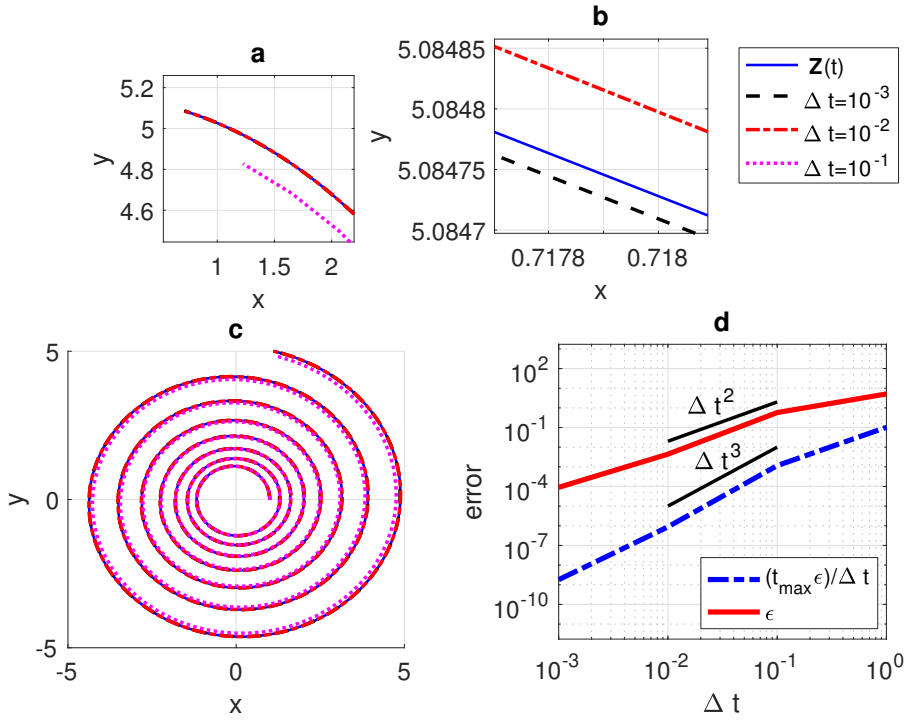
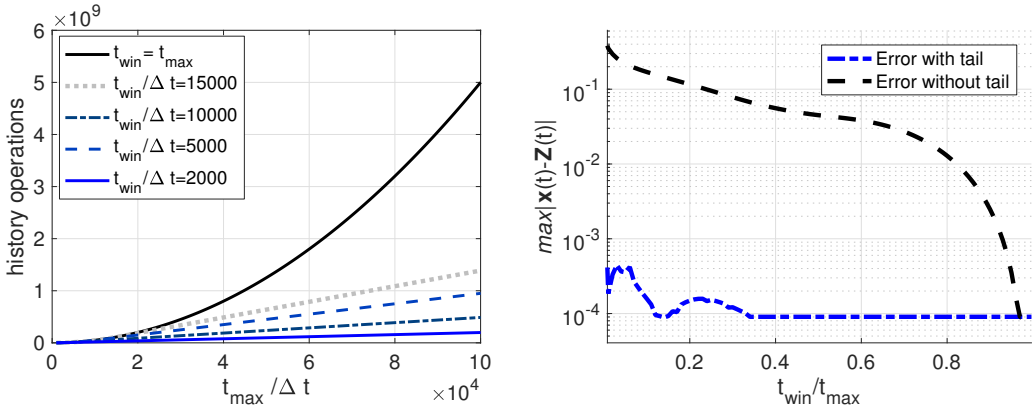


Figure 2.2: *We use an analytical description of inertial particles undergoing rigid body rotation to verify our numerics. In (c) Candeliers analytic solution is cross plotted with our numerics for three step sizes:* $\Delta t = 0.1,\ 0.01$ *and* 0.001*. Zoomed in views are displayed in (a) and (b) to show the differences between the trajectories. In d) we see how the global error,* $\epsilon = \max \|(\mathbf{Z}(t) - \mathbf{x}(t)\|$*, and error per time-step,* $(\Delta t/t_{max})\max \|(\mathbf{Z}(t) - \mathbf{x}(t)\|$ *scale with* Δt*. As expected the method is second order accurate. The fluid velocity flow field was given by* $U_r(\mathbf{x}) = \mathbf{x} \times \omega$ *with* $\omega = [0,0,1]$*,* $\rho f = 1$*,* $\rho p = 1.5$ *and* $r_p = 5 \times 10^{-3}$*. The simulation time was 50 seconds.*

the square of the number of time steps, whilst the window method, coupled with Hinsberg's exponential kernel, grows linearly. Every time step, the cost of evaluating the inner product $c_i g_i$, $i = 1,2,3,...N$ grows to reflect the number of historical steps, N. In the Nth time step the total computational cost is proportional to $\sum_{i=1}^{N} i\beta$, where we have assumed that the cost of evaluating one element of the inner product is a constant β. Therefore, the total computational cost scales as $\beta N(N+1)/2$.Clearly for computational efficiency, we should minimise t_{win}. However, reducing it too far compromises accuracy.

Figure 2.3b demonstrates how the length of t_{win} influences integration error when compared to Candelier's analytic solution. Introduction of the tail approximation significantly reduces the error of the method even for short t_{win}. Maximum accuracy of the method is achieved for a window length that is 1/5 of the length of the simulation, the global error of the method does not improve for further increases in t_{win}.



(a) *The number of operations required to integrate the history window as a function of simulation time.*

(b) *Global error with and without the exponential tail.*

Figure 2.3: *Including the exponential tail approximation improves accuracy and efficiency. In (a), we show that when the tail is included, computational complexity increases linearly with simulation time; when the tail is not used, the total number of operations required to integrate the history force increases non-linearly. Each line shows the number of operations required to integrate the history window for different window lengths, $T_{win} \in [2000, 10^5]$. Shorter window lengths considerably reduce algorithmic complexity. In (b) we show the global error, $max||(\mathbf{Z}(t) - \mathbf{x}(t)||$, for increasing values of T_{win}. The simulation time was 10 seconds ($t_{max} = 10$) and $\Delta t = 10^{-3}$. Including the exponential approximation reduces the error by four orders of magnitude for the same value of T_{win}.*

2.3.1 Comparison with Experimental data

Our numerical scheme compares well with the analytical solution obtained in [60]. To provide a physical validation we also compare our numerical method against experimental data obtained by Esperance [3]. Esperance's experiment consisted of a horizontally oscillating box filled with fluid of density 1840kg/m^3. A sphere was suspended from a light copper wire in the centre of the box so that the influence of gravity could be ignored.

For an oscillating box, the fluid velocity field is given by,

$$U_\omega = A\cos(\Omega t) + Ai\sin(\Omega t) \tag{2.42}$$

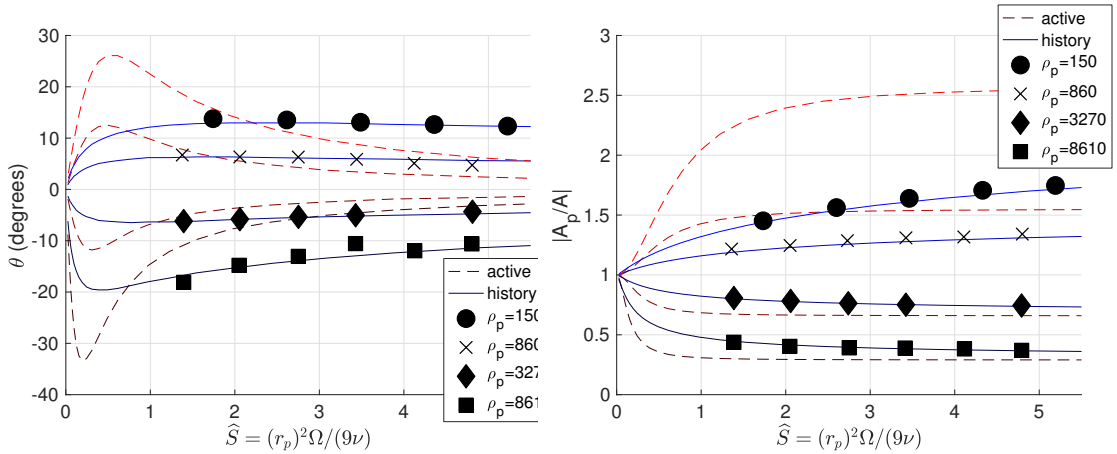
where A and ω are the amplitude and frequency of the oscillations of the cell respectively. In such a flow, inertial particles trace a different path to the fluid in the box. A phase difference between the particles and the box arises, which we shall denote θ. Amplitudes also vary between the fluid and the sphere. We shall denote the amplitude ratio between particle and fluid oscillations with A_p/A, where A_p is the amplitude of particle oscillations.

Esperance used a series of spheres with different material properties, from bubbles to aerosols, to produce a range of density ratios. Each type of sphere had one set diameter. In an oscillating

material	$r_p(mm)$	$\rho_p \mathrm{kg/m}^3$	R
styrofoam	2.23 ± 0.01	140	1.7358
polypropylene	1.98 ± 0.01	860	1.0337
Silicon nitride	1.98 ± 0.01	3270	0.4391
Brass	1.98 ± 0.01	8610	0.1931

Table 2.2: *The particle parameters used in Esperance's experiment.*

flow, a variant on the size parameter is used, given by $\widehat{S} = r_p^2 \Omega/(9\mu)$. To produce a range of $\widetilde{S}$, Ω was changed in the range 20hz to 70hz. Parameter values for the particles are given in table 2.2.



(a) *The phase difference between the oscillations of a box and a sphere*

(b) *the amplitude of oscillations of a sphere divided by the oscillations of a box.*

Figure 2.4: *We present a comparison between simulations using the Maxey-Riley equation and experimental results taken from [3]. The experiment consisted of an oscillating box in which a sphere was suspended. Amplitude and phase differences arise between the particle and fluid oscillations. Blue lines represent simulations including history and the red lines represent simulations using the active tracer. Esperance's experimental results are represented by the points [3].*

Figure 2.4 shows our numerical results cross plotted with the experimental results from Esperance. Figure 2.4(a) shows how the phase difference ϕ varies with $\widetilde{S}$ and R. Figure 2.4(b) shows how the amplitude ratio A_p/A varies over the same range of parameters. We find excellent agreement between our numerics and the experimental data when history is included in simulations (blue lines). When history is neglected, trajectories are materially different (red line). Clearly, history is essential to get good agreement between simulation and reality in this instance.

It has been found in the literature that the influence of history is larger in cases of low density ratio [7, 31]. The largest density ratio used here was relatively low ($\max(\rho_p/\rho_f)$ is $O(10)$); it is expected that history is influential. For larger density ratios, history is expected to have a lesser effect.

2.4 When should each Equation be used?

Armed with a stable numerical method for the solution of the Maxey Riley equation, we now aim to understand the qualitative differences between the trajectories described by the active tracer 2.10, the inertial equation 2.12 and the full Maxey-Riley 2.6. In some parameter regimes, particle dynamics described by the three equations of motion are very similar, in others they differ greatly. To demonstrate such differences qualitatively, trajectories are simulated numerically within the 2-dimensional vortical flow,

$$\mathbf{u}(\mathbf{x}) = \begin{bmatrix} A\cos(y) + B\sin(y) \\ B\cos(x) + A\sin(x)) \end{bmatrix} \tag{2.43}$$

where A and B are flow constants that govern vortex symmetry and fluid velocity. Equation 4.30 defines an incompressible velocity field containing two vortices; it gives a good example of the qualitative differences between the particle descriptions in different parameter regimes. We chose $A = 1$ and B=$\sqrt{2}$ for the flow parameters to produce an asymmetric flow. With these values, the characteristic velocity of the flow $U = 1.66$m/s. The length scale, L, is equal to 2π, because the flow is periodic in squares of 2π by 2π and the characteristic time scale T is equal to $L/Uo = 3.78$s.

In the velocity field described by equation 2.43, we simulate trajectories with parameter values from four distinct regions of the parameter space: two correspond to bubbles, their density is smaller than the surrounding fluid, and two represent aerosols, where the particle density is greater than that of the surrounding fluid. Vortices behave like centrifuges, which cause aerosols to migrate to the edge of a vortex. Conversely, bubbles migrate towards the centre of vortices. Trajectories are plotted in figure 2.5 for the passive, active, and history tracers. Trajectories are also plotted using the inertial equation. Radial migration of particles can clearly be seen in each case, apart from the passive tracer, which follows fluid streamlines.

For dense aerosols, representative of sand in air, similar trajectories are traced by each of equations (2.10), (2.6) and (2.12). For less dense aerosols, for example sand in water, the trajectory including history migrates towards the edge of the vortex more slowly than trajectories described by the active tracer and the inertial equation. For bubbles, the effects are more pronounced still: including history significantly reduces vortex migration.

In each case in figure 2.5 the parameter R/S is fixed at 10, but the differences between trajectories described by each of the four equations of motion are clear. We can see that changing the density ratio, R, and particle size parameter, S, produces different behaviours for each of the equations of motion. The vortex ejection rate is not constant for constant R/S, but scales with a more complicated function of both parameters.

It is interesting to see how these two parameters affect the behaviour of a particle when they are considered separately. In figure 2.6 particles were initialised again from the top of a vortex and trajectories were simulated using the full MR equation in (2.6). In figure 2.6a) trajectories have been simulated with size parameter ranging from 1.5×10^{-6} to 1.5×10^{-4}. In

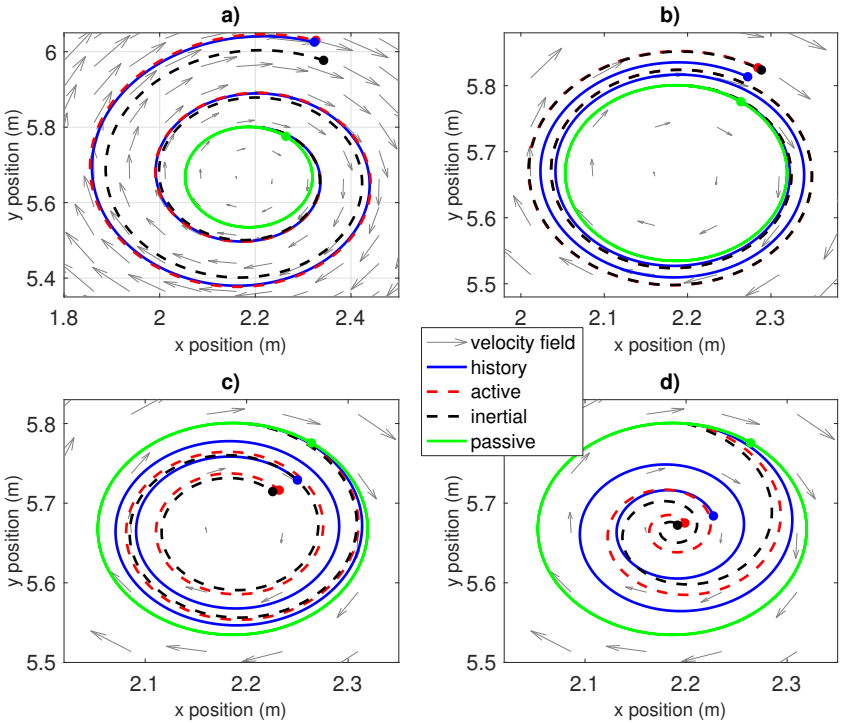


Figure 2.5: *Each of the four versions of the equation of motion are used to calculate a trajectory in the flow given by equation 2.43. The particle was released at the top of a vortex with initial velocity equal to the local undisturbed fluid velocity. Each of the four subplots represent behaviour in different parameter regimes: a) a very dense aerosol - sand in air with* $R = 3 \times 10^{-4}$ *and* $S = 2.96 \times 10^{-5}$*; b) a lighter aerosol, sand in water with* $R = 0.4568$ *and* $S = 0.0542$*; c) a bubble half as dense as the fluid it is advected in with* $R = 1.084$ *and* $S = 0.023$*; d) a bubble that is many times less dense than the surrounding fluid with* $R = 1.997$ *and* $S = 0.099$*,* $St = 0.057$*.*

Figure (2.6b) trajectories have been simulated with R ranging from 0.095 to 2. Both R and S influence the rate of divergence from the underlying fluid velocity. A fixed S and varied R can produce dynamics ranging from rapid migration towards the centre of a vortex to rapid ejection. The density parameter, R, influences both the rate, and direction of radial migration. The direction of migration can either be towards the centre of rotational flow features or towards the edge. Varying S while keeping R constant changes only the vortex migration rate, but not the direction.

Having established the qualitative differences between the approximations to the equations of motion, we now attempt to quantify the differences in an idealised vortex filled flow. From figure 2.2 it can be seen that, for a time step of $\Delta t = 10^{-3}$, the error per time-step of the numerical method is $O(10^{-8})$. When this is multiplied by the number of time steps in one characteristic time period, namely 3786, we arrive at the expected global error of the simulation. If the difference between trajectories exceeds the expected global error, we assume they are materially different

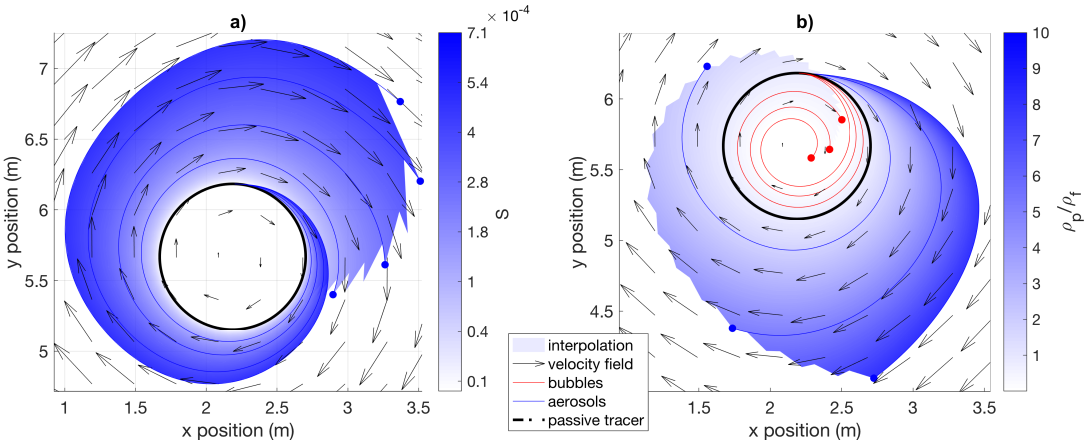


Figure 2.6: *Both the particle diameter and density ratio influence the behaviour of particle trajectories around a vortex. Trajectories described by the full MR equation in equation 2.6 are released from the top of the vortex at the velocity of the fluid. The underlying fluid velocity is described by equation 4.30. In a), the particle diameter is increased leading to rapidly increasingly vortex ejection rates. In b), we vary the density parameter R, which affects both the rate and direction of radial migration of particles.*

trajectories influenced by our modelling choices rather than numerical errors.

Using the error threshold of $\epsilon = 1.154 \times 10^{-5} = 3786 \times 10^{-8}$, we now explore the parameter space for regions in which model refinement offers significant improvement. The parameter space is defined by (ρ_p/ρ_f) and S and is populated by a grid of points at discrete intervals in each dimension. For each grid point $[S_i,(\rho_p/\rho_f)_j]$, a simulation is run for one characteristic timescale in the fluid velocity field in equation 4.30. We recall that the characteristic timescale is $T = U/L$.

We treat the problem as though we were investigating the error produced by a numerical method. In such cases, the maximum difference between an analytic solution and the numerical approximation are often taken as the error of the method. The base assumption is that the full Maxey-Riley equation is the best approximation to the behaviour of inertial particles on offer. As such, the discrepancies of the various approximations will be made with reference to this relative truth. For each model, 50 trajectories are released from every position in the parameter space defined by $[S_i,(\rho_p/\rho_f)_j]$. We calculate the maximum Euclidian distance between each model pair simulated with parameters from each position in the parameter space. The average is then taken over each of the 50 initial positions in the flow. For every position in the parameter space, we now have a difference between each of the models. We plot the results in figure 2.7.

Each subplot in figure 2.7, represents a colour map plotted over the parameter space described above. Figure (2.7a) shows the maximum distance between the passive and active tracers defined as, $d1 = \max(|passive - active|, T)$. Similarly, figures (2.7b) and (2.7d) show $\max(|passive - inertial, T)$, and $\max(|history - active|, T)$ respectively. Figure (2.7c) shows how the particle Reynolds number, Re_p varies over the same parameter space. On each of these subplots are a

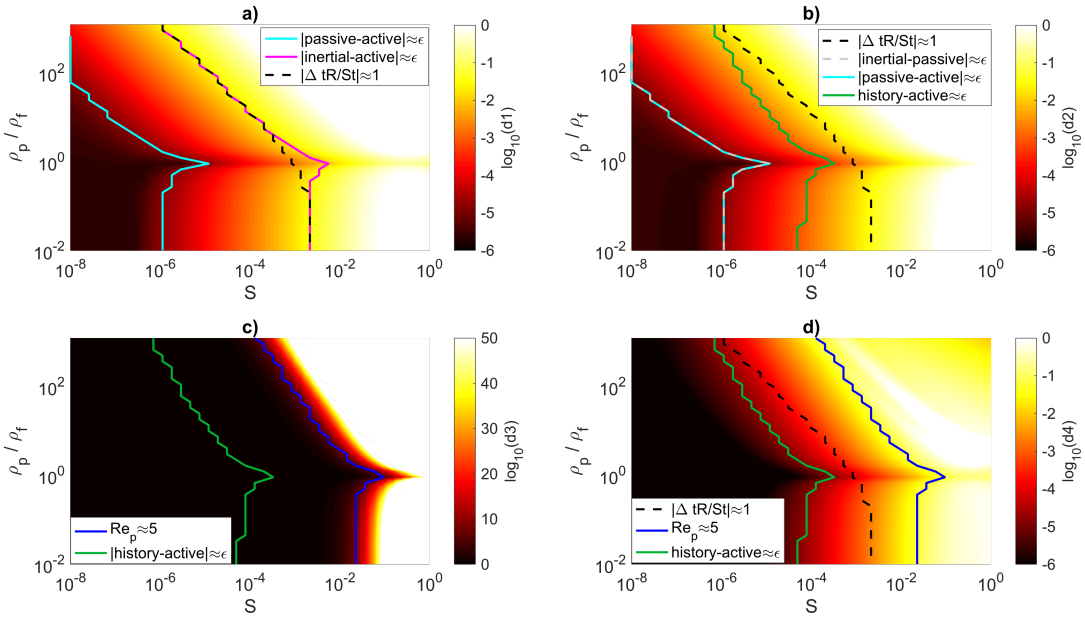


Figure 2.7: *We compare each of the models with one another across a well resolved parameter space defined by the size parameter S and the density ratio, ρ_p/ρ_f. In a) we plot* $\log_{10}$ *of the maximum displacement between the active and passive tracer as a colour map. In b), we show the displacement between the inertial equation and the passive tracer and in d), we show the displacement between the active tracer and the history tracer. In c) we show the particle Reynolds number as a function of S and ρ_p/ρ_f. The region between the blue and green lines indicates the region for which history should be included.*

series of lines defining boundaries between different regions of interest in the parameter space. The first of these, plotted in cyan on figures (2.7a) and (2.7b), shows where the difference between the passive and active tracers exceeds the numerical error of the method. The dashed grey line plotted on figure (2.7b) indicates where the difference between the inertial equation and passive tracer exceeds the model accuracy. These lines mark the location in the parameter space where inertial effects exceed numerical errors.

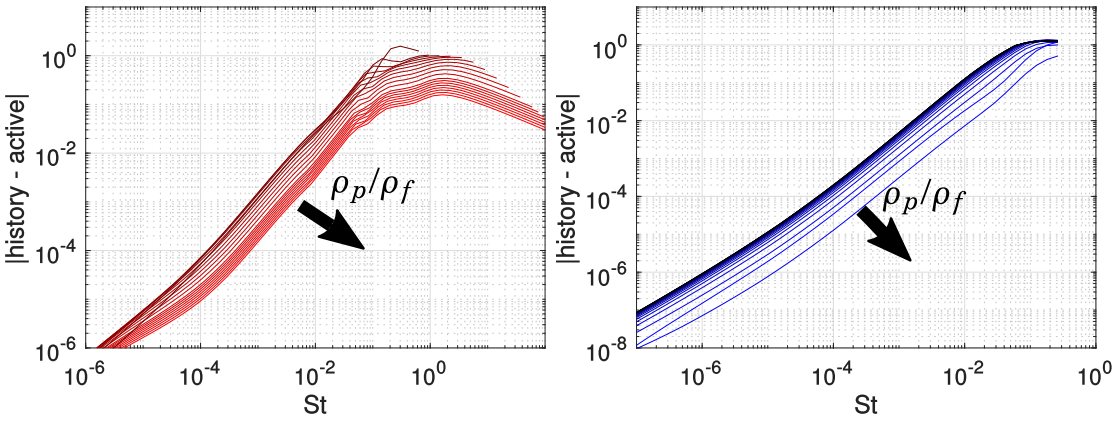
The dashed black line, plotted on Figures (2.7a), (2.7b) and (2.7d), shows where equation 2.30 loses stability for a time step of 10^{-3}. A trajectory would be numerically unstable if it were simulated using equation 2.30 from a position in the parameter space with smaller size parameter than this line. The magenta line plotted on figure (2.7a) indicates where the inertial equation begins to differ significantly form the active tracer. This is very similar to where the stability of equation 2.30 is lost.

The green line plotted on figures (2.7b), (2.7c) and (2.7d) marks the boundary in the parameter space for which the influence of history exceeds numerical errors. Finally, regions where the size parameter is greater than the blue line in Figures (2.7b) and (2.7c) are regions where the particle

Reynolds number exceeds 5. We assume the Maxey-Riley equation to be invalid for larger particle Reynolds numbers. We can find an approximate expression for the line of blue line for the case of aerosol particles. Each line in figure 2.7 approximately satisfies a power law in S. For aerosols we find that $\rho_p/\rho_f \approx 0.0045 S^{-1.4038}$. So to ensure the particle Reynolds number does not exceed 5, we must ensure that $\rho_p/\rho_f < 0.0045 S^{-1.4038}$.

An important observation from this analysis is that inappropriately neglecting the history force produces the same numerical errors for all density ratios, it simply occurs for different size parameters. In figure (2.7d), the distance between the green and blue lines remains approximately parallel as a function of ρ_p/ρ_f. Between the two lines the influence of the history force exceeds numerical errors.

Often, the important parameter when investigating the behaviour of inertial particles is the Stokes number, $St = S/(RT)$. Particle clustering, for example, reaches a maximum for $St \approx 1$ in turbulent flows. In figure 2.8, we investigate how the Euclidian distance between the active and history tracers changes with the Stokes number ST. In figure 2.8(a) we see that the influence of history on aerosol particles reduces for increases in the density ratio ρ_p/ρ_f. Bubbles in figure 2.8(b) display the same behaviour. Interestingly, the maximum difference between history and active tracer trajectories occurs for $St \approx 1$ in all cases.



(a) *aerosols* ***(b)*** *bubbles*

Figure 2.8: *History is not equally influential for all density ratios. The smaller the density ratio the more active tracer and history tracer trajectories diverge. Euclidian distances between the trajectories described by each of the models reaches a peak at $St \approx 1$ for all density ratios. For increasing density ratios ρ_p/ρ_f and a constant Stokes number, history is less influential.*

2.5 Conclusions

We have shown that using an explicit method to integrate the MR equation fails for small particles. Although deviations from fluid velocities are limited for small particles, we have shown that

particles display inertial properties below the limit of instability. By using an implicit method such as the BDF2, problems of instability can be circumvented.

Using the window method, coupled with Hinsberg's tail approximation, produces the best trade off between accuracy and numerical efficiency. In fact, no accuracy is lost by using the tail approximation, provided the history window is long enough.

Simulations have been compared both against analytical solutions of the MR equation and against experimental data. Our numerics perform well in both tests, allowing it to be used as a benchmark against which to test further methods in the following chapters.

There is a clear region of the particle parameter space, defined by S and R, for which the history force should be used. The specific combination of S and the density parameter R should be considered when deciding whether or not history is important for a specific application. History is apparently influential on simple particle trajectories for all density ratios. We defined history as being influential when it altered trajectories by more than the magnitude of numerical errors. Of course, exceeding numerical errors for a single trajectory does not necessarily mean that history will be influential on an ensemble of particles. For an ensemble of particles, it is the statistical properties that we are interested in. Subtle deviations from fluid streamlines shown here may not translate into differences in statistical properties, such as the degree of particle clustering or rate of collisions. In later chapters we will investigate particle clustering and collision rates in the presence of history.

We also demonstrated that, for all density ratios, the maximum influence of history occurred at $St \approx 1$. History is less influential for larger density ratios for the same value of St.

CHAPTER 3

A TRANSFER OPERATOR METHOD FOR INERTIAL PARTICLES

3.1 Introduction

In this chapter we explore a method for capturing the ensemble behaviour of inertial particles in a computationally efficient manner. In the first chapter, the dynamics of individual particle trajectories in steady flows were investigated. Whilst such analysis is useful in understanding dynamics on the micro scale, in most real world applications inertial particles are part of larger ensembles of particles with their own macro-scale dynamics. In fact, many real world examples of inertial particle advection involve large numbers of particles that interact both with one another and the surrounding fluid. Attempting to simulate such processes numerically requires large amounts of computing power.

An example of a process that carries a high simulation cost is raindrop formation. In the early stages of raindrop formation billions of particles are advected by the turbulent vortices of the cloud. The number of particles per unit volume of fluid can range from 50 cm^{-3} in ocean clouds, to 1000 cm^{-3} in continental clouds [29]. Newly nucleated droplets are of the order of microns in diameter, but by the time they fall as raindrops, they have increased their diameter to the order of millimetres. The increase in droplet diameter occurs through a number of processes, but in warm clouds a primary mechanism is through droplet collisions [15]: two particles collide and coalesce to form a single larger droplet. Simulating this phenomenon in a cubic metre of cloud, even at the lower end of the scale for spatial density, requires the simulation of 5×10^7 particle trajectories. Between each time step, collisions must be searched for; when undertaken naively, a task that scales as the number of droplets squared. It is clear that a more sophisticated approach to modelling the process is desirable over pairwise collision detection of billions of individual trajectories.

In other examples, the ensemble behaviour, rather than the behaviour of individual particle trajectories, is the important concern. In sediment transport it is of little consequence where an individual grain of sand comes to rest upon a riverbed, but in regions of the flow where fluid velocities are low, increased particle sedimentation leads to significant inhomogeneities in the depth of the river [78]. Similarly, in the transport of a volcanic ash cloud the important factor is the spatial density and trajectory of the ash cloud in general, not the path of a single soot particle [10]. If it is possible to capture the emergent behaviour of the ensemble, caused by the individual inertial properties of the particles, it would lead to significant computational savings.

Our aim is to provide a tool that efficiently captures the clustering behaviour of inertial particles. It is also important that the method can be extended to allow investigation of processes that involve particle interactions or two way coupling between the particles and the fluid. In applications such as raindrop formation, the sedimentation of silt or sand in a river, or the effluence of sewage from a pipe, the inertial properties of the particles play a role. However, to capture the problems in their entirety the forcing on the fluid from the particle phase must be considered. Further, in the application of raindrop formation, the particle collision rate is of specific importance: it is a major control on the growth rate of the particles [15].

Inertial particles appear in a range of contexts. The level of model complexity required to accurately capture particle dynamics is dependent on the appropriate particle parameters for the specific application. Instances when the influence of the Basset force is significant, present a considerable complication to modelling. In the first chapter, the influence of the history force on individual particle trajectories was ascertained for a range of parameter regimes. When history is included, it significantly increases the cost of simulations: both computationally and in terms of complexity. When inclusion of history is not necessary, the door is opened to a wider range of modelling tools; the computational cost, both in terms of memory use and simulation time, is reduced. It is important to understand exactly what effect the Basset force has on the ensemble, to develop better understand of when its inclusion is necessary. We also investigate whether our approach can be augmented to include the influence of history.

We begin our journey by breaking down the fundamental properties of an inertial particle. In many instances, the complicated and nuanced behaviour of a fluid flow can be condensed into a series of vortices. For this reason, we develop our method in a simple flow comprising a series of periodic vortices. Within such a flow, inertial particles cluster either between the vortices, or within the vortex centres, depending on their relative density to the fluid. The general behaviour we wish to capture, is demonstrated in figure 3.1. The figure shows the distribution of inertial particles in a 2-dimensional space at three instances in time: after $2.4T$, $4.8T$ and $7.2T$, where T is a characteristic timescale given by $T = U/L$. We use the root mean square velocity for U and the domain size for L. Particles were advected in a flow containing periodic vortices. As the particles are more dense than the fluid in which they reside, the particles migrate towards the boundaries between the vortices.

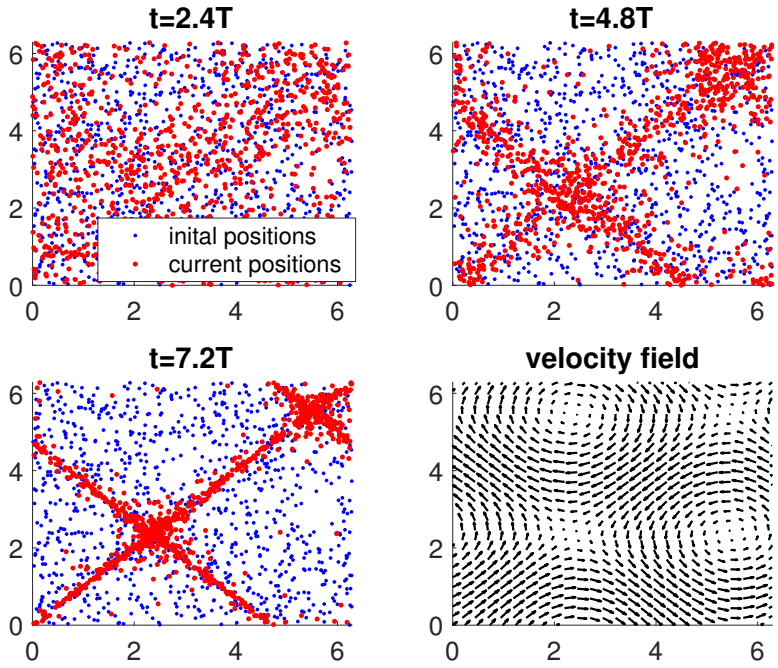


Figure 3.1: *Inertial particles that are more dense than the fluid migrate towards the edge of vortices. Three snapshots of a simulation demonstrating this behaviour are shown for* $\rho_p = 1000$*,* $\rho_f = 1$ *and* $S = 10^{-4}$*. Initial positions of the particles are displayed as small blue points. Particle positions after* 2.4*,* 4.8 *and* 7.2 *characteristic timescales are plotted in red in each subplot. As time progresses, particles move closer to the edges of the vortices; after* 7.2 *characteristic timescales the particles line up along the separatrices.*

It is the ensemble radial migration towards the edges of the vortices, shown in figure 3.1, that any method we develop must capture. Without requiring the simulation of each individual trajectory, we desire a continuous representation, both in space and time, of the evolution of the particle concentration field. This particular goal provides a series of advantages. Because the method assumes the continuum limit of particle concentration, increasing the represented number of particles has no increase on computational cost. Information is provided about the whole flow, not just the region surrounding a trajectory, which allows the consideration of the feedback between the fluid phase and particle phase. This feedback becomes increasingly important as the concentration of the particles increases [22].

One potential solution to our problem is to simulate large numbers of particles and bin their spatial locations throughout the simulation; providing information about the particle concentration field throughout the flow. However, spatial binning of particles provides spatially biased information about the particle concentration field, because regions where particles are unlikely to go are sparsely sampled.

We desire a continuous representation of particle concentration that retains the inertial

properties of the particles. In the following, we investigate using a transfer operator method for inertial particles to track the particle concentration field. In the lecture notes from Omri Sarig on the "introduction to the transfer operator method" [79], is a simple thought experiment. "*Can you predict where individual ink particles will end after one minute? NO: the motion of ink particles is chaotic. 2. Can you predict the density of the ink particles after one minute? YES: it will be nearly constant, equal to* |*mass of ink*| / |*volume of water+ink*|.". Whilst it is not our desire to describe such a simple mixing problem, the example points out that the least predictable part of the system is of little consequence to the general macro scale behaviour of the system. When numerically simulating the trajectories of particles in a chaotic system, the final locations of the particles can be plagued by errors: small deviations in their trajectory imposed by numerical errors results in particles ending up in vastly different locations. It is possible however, to predict the spatial density of the particle ensemble and to quantify the errors on its prediction.

Consider the active tracer description of an inertial particle,

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t)) = \begin{bmatrix} \mathbf{v}(t) \\ \frac{3}{2}R\frac{D\mathbf{u}(\mathbf{y},t)}{Dt} - R/S(\mathbf{v}(t) - \mathbf{u}(\mathbf{y},t)), \end{bmatrix} \tag{3.1}$$

where R is the density parameter $\frac{2\rho_f}{(2\rho_p+\rho_f)}$ and S is the size parameter $\frac{2r_p^2 Re}{9L^2}$, ρ_p and ρ_f are the particle and fluid material densities respectively and r_p is the radius of the particle. The position of a particle at time t is given by $\mathbf{y}(t)$ and $\mathbf{v}(t)$ is their velocity. The undisturbed fluid velocity field in the absence of a particle is given by $\mathbf{u}(\mathbf{y},t)$. We define a Reynolds number Re as $LU\rho_f/\mu$, where L and U are a characteristic length scale and velocity of the flow respectively and μ is the dynamic viscosity of the fluid.

Equation 3.1 can be integrated to move an initial particle position, $\mathbf{x}_0 = [\mathbf{y}_0, \mathbf{v}_0]$ to give a new position in phase space, $\mathbf{x}(t)$, after some time t. The active tracer is a function that takes an individual particle position, and moves it in phase space. We denote the phase space of the active tracer system as, $\mathscr{M} \in \mathbb{R}^d$, where d is the number of dimensions. Imagine that instead of a way of moving individual points in $\mathscr{M}$, we want to operate on a continuous function over $\mathscr{M}$. Instead of evolving the trajectories themselves, we act on the spatial density of trajectories, $\rho(\mathbf{x},t)$.

We consider that $\mathscr{M}$ is partitioned into infinitesimal subsets, $\mathscr{M}_i \subset \mathscr{M}$, each with a probability of containing a particle [80]. The probabilities across the subsets together form the probability density function $\rho(\mathbf{x},0)$. We want a way of mapping some initial probability distribution of particles, $\rho(\mathbf{x},0)$, to some new distribution $\rho(\mathbf{x},t)$ at time t. Of course, the action on $\rho(\mathbf{x},0)$ must obey the dynamics of equation 3.1 and the process must be conservative so that $\int_{\mathscr{M}} \rho(\mathbf{x}_0,0)d\mathbf{x} = \int_{\mathscr{M}} \rho(\mathbf{x},t)d\mathbf{x}$.

A transfer operator is exactly what we seek. It evolves a scalar field in time, and is defined over any arbitrary phase space. Often referred to as the Perron-Frobenius operator, Φ^t can be thought of as an infinite dimensional transition matrix which encodes the flow of particles between the infinitesimal partitions of $\mathscr{M}_i \subset \mathscr{M}$. Following [80], the transfer operator, Φ^t, can be

defined as,

$$\rho(\mathbf{x},t) = \Phi^t \rho(\mathbf{x}_0,0) = \int_{\mathscr{M}} \delta(\mathbf{x} - g^t(\mathbf{x}_0))\rho(\mathbf{x}_0,0)d\mathbf{x}_0, \tag{3.2}$$

where in our case $g^t(\mathbf{x}_0)$ is the solution of the active tracer system and δ is the Dirac delta [80]. To acquire a more physical understanding of the definition, equation 3.3 can be transformed through the substitution $g^{-t}(\mathbf{y}) = \mathbf{x}_0$ to give,

$$\begin{aligned}
\rho(\mathbf{x},t) &= \int_{\mathscr{M}} \delta\left(\mathbf{x} - g^t g^{-t}(\mathbf{y})\right) \rho(g^{-t}(\mathbf{y}),0) dg^{-t}(\mathbf{y}) \\
&= \int_{\mathscr{M}} \delta(\mathbf{x}-\mathbf{y})\rho(g^{-t}(\mathbf{y}),0)\left|\det\left[\nabla g^{-t}(\mathbf{y})\right]\right| d\mathbf{y} \\
&= \rho(g^{-t}(\mathbf{x}),0)\left|\det\left[\nabla g^{-t}(\mathbf{x})\right]\right| \\
&\implies \frac{\rho(\mathbf{x},t)}{\left|\det\left[\nabla g^{-t}(\mathbf{x})\right]\right|} = \rho(\mathbf{x}_0,0),
\end{aligned} \tag{3.3}$$

which tells us that the new density in a location is dependent on the change in volume of that location. If the size of the determinant of the Jacobian is less than one, then the region has contracted and the density has increased, if the determinant of the Jacobian is greater than one, the region has expanded and the density reduced.

A similar result can be obtained by considering conservation of trajectories. Because the same number of trajectories exists in $g^t(\mathscr{M}_i)$ as in $\mathscr{M}_i$, then,

$$\int_{\mathscr{M}_i} \rho(\mathbf{x}_0,0)d\mathbf{x}_0 = \int_{g^t(\mathscr{M}_i)} \rho(\mathbf{x},t)d\mathbf{x} \tag{3.4}$$

which by making the substitution $\mathbf{x} = g^t(\mathbf{x}_0)$ into the integral on the R.H.S. gives,

$$\begin{aligned}
\int_{\mathscr{M}_i} \rho(\mathbf{x}_0,0)d\mathbf{x}_0 &= \int_{\mathscr{M}_i} \rho(g^t(\mathbf{x}_0),t)dg^t(\mathbf{x}_0) \\
&= \int_{\mathscr{M}_i} \rho(g^t(\mathbf{x}_0),t)\left|\det\left[\nabla g^t(\mathbf{x}_0)\right]\right| d\mathbf{x}_0 \\
&\implies \frac{\rho(\mathbf{x}_0,0)}{\left|\det\left[\nabla g^t(\mathbf{x}_0)\right]\right|} = \rho(\mathbf{x},t)
\end{aligned} \tag{3.5}$$

which is equivalent to equation 3.3 forwards in time.

To summarise, the transfer operator Φ^t, takes the initial probability density function of particle positions $\rho(\mathbf{x}_0,0)$, and maps it to a new state at time $\rho(\mathbf{x},t)$. The mapping can be based on an arbitrary dynamical system, but in our case $g_t(x_0)$ is generated by the active tracer system in equation 3.1. We seek a transfer operator for inertial particles because a transfer operator acts on the whole particle phase space simultaneously, providing particle information everywhere in the flow. Where the feedback on the fluid phase is important an Eulerian approach is vital, providing significant advantages over Lagrangian approaches which only provide information

around particle trajectories. Further, as the transfer operator acts on the PDF of particle positions, $\rho(x,t)$, a transfer operator can be used to represent any arbitrary number of particles with no increase in complexity. The transfer operator evolves $\rho(x,t)$ such that the attractive regions of the space will emerge.

In the following we will be dealing with higher dimensional spaces. To help with explanations we will use the term configuration space to mean the physical space occupied by particle positions. Phase space denotes the space encompassing both the position and velocity of the particles, and velocity space will be used to refer to just the velocity dimensions.

3.2 Construction of the transfer operator

We now turn our thinking to the discretisation of a transfer operator in an arbitrarily high dimensional space. The transfer operator, $\Phi^t : \mathbb{R}^d \to \mathbb{R}^d$, is an infinite dimensional operator acting on a d dimensional phase space. We seek a finite dimensional linear operator, $W_{ij} : \mathbb{R}^d \to \mathbb{R}^d$ which is a discretisation of Φ^t [81]. An appropriate discretisation will take an infinite dimensional operator, Φ^t, which acts on a continuous space, and create a discrete, numerically realisable operator.

Commonly, a transfer operator is discretised using Ulam's method [80–83]. Recall that Φ^t acts on the spatial density of trajectories $\rho(\mathbf{x},t)$ in the particle phase space, $\mathscr{M}$, which is partitioned into infinite subsets, given by $\mathscr{M}_i \subset \mathscr{M}$. The crux of Ulam's method is that instead of dividing $\mathscr{M}$ into infinitely many subsets, it is discretised into a finite set of subsets. We assume that the particle density is uniform within each subset, producing a discrete probability density function of particle positions in phase space, P_j. Ulam's method produces an operator W_{ij} that maps P_j^n to P_j^{n+1}, where n is the current time step and j is the cell index of the discretised space.

By considering the indicator function $\mathbb{1}_{\mathscr{M}_i(\mathbf{x})}$,

$$\mathbb{1}_{\mathscr{M}_i(\mathbf{x})} = \begin{bmatrix} 1, & \text{if} \quad \mathbf{x} \in \mathscr{M}_i \\ 0, & \text{otherwise} \end{bmatrix}, \tag{3.6}$$

we can rewrite equation 3.3 in the discrete form [80, 81, 83],

$$P_i^{n+1} = P_j^n \int_{\mathscr{M}_i} \int_{\mathscr{M}_j} \mathbb{1}_{g^{-1}(\mathscr{M}_j)} \cdot \mathbb{1}_{\mathscr{M}_i} d\mathbf{x} d\mathbf{x}_0, \tag{3.7}$$

From equation 3.7, we can see that the form of a discrete transition matrix $\mathbf{W}_{ij}$ is,

$$W_{ij} = \frac{|\mathscr{M}_i \cap g^{-1}(\mathscr{M}_j)|}{|\mathscr{M}_i|}, \tag{3.8}$$

where $|\cdot|$ denotes the volume of a set. With a matrix formulation of the Perron-Frobenius operator we are able to write,

$$P_i^{n+1} = \mathbf{W}_{ij} P_j^n. \tag{3.9}$$

A good way of visualising the idea is to consider each of the subsets, (which we will now refer to as cells) as a node in a graph, and W_{ij} as the matrix of weights linking the nodes. The operator, W_{ij}, is constructed based on the flux of P_j through the faces of the cells.

Figure 3.2 gives an example of a graph over a 2-dimensional Cartesian grid with weights between nodes representing fluxes through the faces of cells. The weights are displayed as numbers adjacent to their corresponding red flux arrow and take arbitrary values to illustrate the idea. There is a flux of P_j^n through every face indicated by a weight. The picture changes from one of a uniform particle distribution to one illustrating the location of attractors.

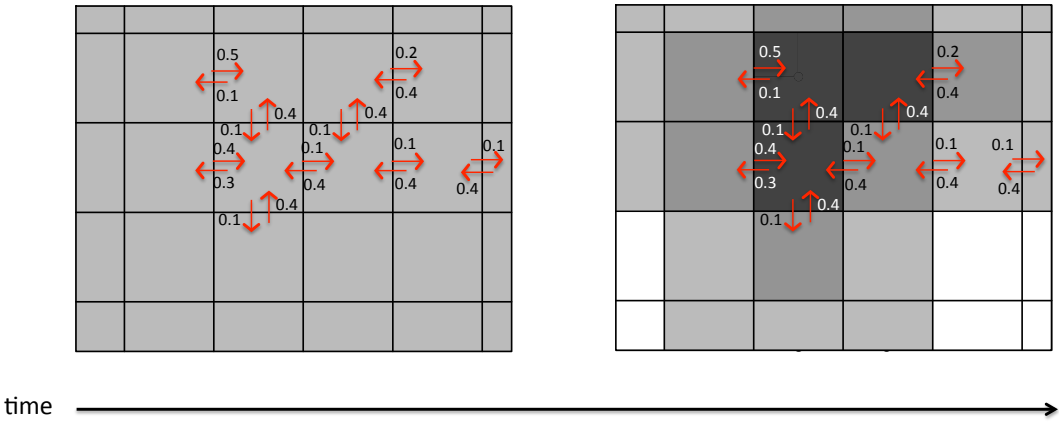


Figure 3.2: *A discretised transfer operator encodes the flow of the probability distribution of particle positions between cells in a discretised space. Red arrows indicate the direction of the fluxes through cell faces and edge weights represent the proportion of the origin cell that is transferred through the indicated face. Although the dimensionality of the space is arbitrary, the diagram shows an example in a 2-dimensional space for simplicity.*

The projection of Φ from an infinite dimensional operator onto a finite dimensional operator W_{ij} must satisfy certain conditions. The method for calculating the values in W_{ij} must be simple enough to be tractable in higher dimensions. It must also be conservative: globally the number of particles in the system must not change. This is achieved by restricting the sum of each column of W_{ij} to 1. Further, W_{ij} must be a stochastic matrix. In a stochastic matrix all matrix entries are non-negative real numbers between 0 and 1.

We can see from equation 3.8, that W_{ij} is the proportion of trajectories that originated in cell $\mathscr{M}_j$ and arrived in cell $\mathscr{M}_i$ based on the pre-image of the dynamical system. Each cell, $\mathscr{M}_j$, can be thought of as representing a host of initial particle positions in phase space, which are evolved in time. Final particle positions that intersect cell $\mathscr{M}_i$, indicate the movement of density from cell j to cell i. Generally, the evolution of trajectories is conducted backwards in time. However, inertial particle trajectories are unstable backwards in time as they diverge exponentially from an inertial manifold [4]. Therefore, we evolve particle trajectories forwards in time from their initial positions in phase space. The final positions of the particles are distributed over the neighbours of

the cell from which they originate. The proportions with which trajectories are divided amongst the neighbourhood of their originating cell defines the weights of the edges linking the nodes. For example, if 50% of particles arrive in the cell to the right of their original cell, then the weight between those cells would be 0.5. The calculated weights represent the probability of a particle from one cell, arriving in another cell after time Δt.

It is commonly suggested that the weights linking the cells can be found as follows: a number of particle trajectories are initialised from within each cell; after the particles are advected by the fluid for some time, Δt, the number of particles residing in each cell is recorded [81, 84, 85]. The proportion of particles residing in each cell, after the short advection, denotes the weight of the link between the original cell and each of its neighbours. Figure 3.3 illuminates the remapping process: the blue points represent the original positions of the particles and the red points represent their positions after Δt seconds; The weights have been calculated and displayed as fractions in the corners of their respective cells. Each weight represents the link from the central cell to the cell containing the weight in its corner.

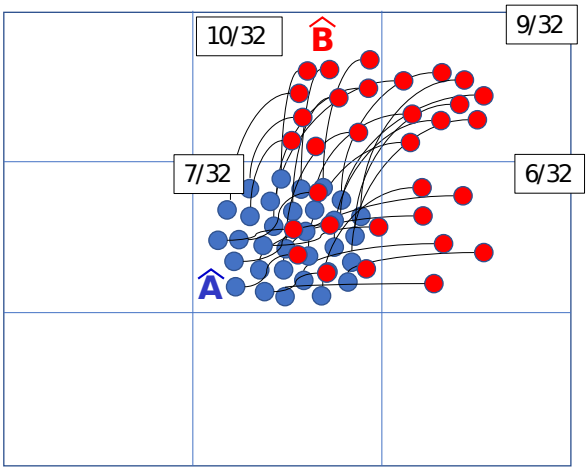


Figure 3.3: *A common approach for the construction of a discrete transfer operator is to first initialise particles from the blue positions labelled $\widehat{A}$. After a short integration period Δt, the particles occupy the new red positions labelled $\widehat{B}$. The proportion of the red positions in each new cell, denotes the weights on each of the edges leaving the original cell. The weights are denoted by the fractions in rectangular boxes.*

Counting the particles in each cell has advantages: particles can end up in any cell and the corresponding flux can be readily incorporated in the transfer operator; it is simple to implement; and sub-grid level flow features can be captured (provided they are resolved by the flow field $\mathbf{u}(x,t)$). A problem with the method is that its accuracy is not only dependent on the grid resolution, but also on the number of particles simulated from within each cell. In a higher dimensional implementation, where the number of cells is large, increasing the number of particles is computationally costly. Instead, we propose a method in which the accuracy will

depend on higher order methods instead of larger numbers of trajectories.

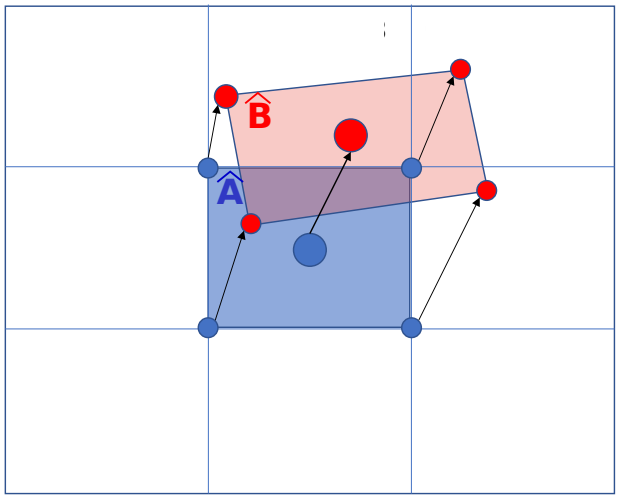


Figure 3.4: *In our approach for calculating the transfer operator, particle trajectories are evolved from the corners of grid cells. After a short integration period, Δt, the particles define a new shape, $\widehat{B}$. The new shape, $\widehat{B}$, is projected onto the original grid and distributed according to the grid cells it overlaps.*

In our alternative method, which we define the inertial remapping method (IRM), the weights on the edges are realised by evolving trajectories from the corners of each cell. After a short integration in time, the particles define a new shape overlapping the old grid. The original shape defined by the grid is denoted by $\widehat{A}$, and the new shape, defined by the final positions of the particles, is denoted by $\widehat{B}$. The locations of the particles now define the new shape, $\widehat{B}$, overlapping several of the original grid squares. The particles from within the original grid square are redistributed based on the proportion of the new shape that overlaps each of the neighbouring grid squares. Figure 3.4 gives a two dimensional example of the IRM process. A trajectory from each corner of $\widehat{A}$ is evolved forward in time for Δt seconds to create the new shape $\widehat{B}$, which will be redistributed over the cells beneath it.

A simple first order mechanism is utilised to calculate the proportion of shape $\widehat{B}$ that overlaps each of the cells beneath it. We first calculate the centroid of shape $\widehat{B}$ and denote it with $\widehat{\mathbf{B}}_c$. The centroid of shape $\widehat{A}$ we denote with $\widehat{\mathbf{A}}_c$. We now take the difference between the centroids of shapes $\widehat{A}$ and $\widehat{B}$, and denote it with the vector $\mathbf{x_c} = \widehat{B}_c - \widehat{A}_c$. Next $\mathbf{x_c}$ is divided by Δx, where Δx is the grid spacing, to find the proportion of a cell for which the particles have moved. The process is repeated for each dimension separately, and the product of the results from each dimension, give one element in the jth column of W_{ij}.

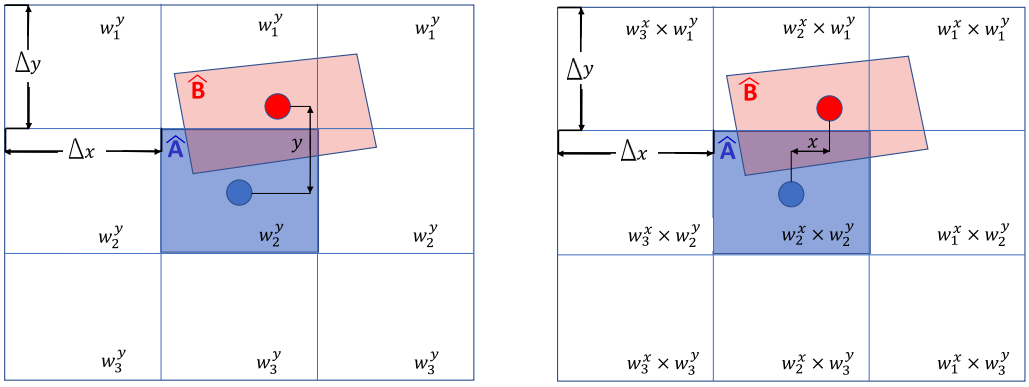
The algorithm for the calculation of the columns of W_{ij}, in a phase space discretised into

N cells, is given in equation 3.10. For each dimension the calculation depends on whether the translation of the centroid has been positive or negative. We make use of the heaviside step function, $H(x)$, to incorporate the direction dependent element of the algorithm. We iterate through the cells as follows:

$$
\begin{aligned}
&\text{for } i = 1,2,3...N \\
&\qquad \text{if } i \text{ exists in the first order neighbourhood of cell } j \\
&\qquad \text{then} \\
&\qquad\quad x_c^d = \mathbf{B}_c^d - \mathbf{A}_c^d \text{ where } d = [1..., \text{number of dimensions}] \\
&\qquad\quad w_{\mathbf{1}}^d = H\left(-\frac{x_c^d}{\Delta x^d}\right)\left|\frac{x^d}{\Delta x^d}\right| \\
&\qquad\quad w_{\mathbf{2}}^d = 1 - \left|\frac{x_c^d}{\Delta x^d}\right| \\
&\qquad\quad w_{\mathbf{3}}^d = H\left(\frac{x_c^d}{\Delta x^d}\right)\left|\frac{x^d}{\Delta x^d}\right| \\
&\qquad\quad W_{ij} = \prod_d w_i^d . \\
&\qquad \text{else} \\
&\qquad\quad W_{ij} = 0
\end{aligned}
\tag{3.10}
$$

, where $|\cdot|$ denotes the absolute value. The algorithm loops through the cells in P_j. We are calculating how each cell in P_j is distributed amongst its order 1 neighbourhood. The diagram in figure 3.5 demonstrates how the algorithm in equation 3.10 works in 2 dimensions. Originally, particles on the corners of the central cell form shape $\widehat{A}$. We will calculate how shape $\widehat{A}$ is distributed amongst its surrounding cells to create a set of weight components, w_k^d. In this case, we have two dimensions, so d can take two values: x, for horizontal components and y, for vertical components. For each dimension, the central cell has three neighbours: one in the negative direction, one in the positive direction, and itself. Each of these directions is represented by k: $k = 1$ indicates the row or column either above, or to the right of the central cell; $k = 2$ denotes the row or column containing the central cell; and $k = 3$ denotes the row or column of cells to the left, or below, the central cell. Because we have two dimensions, there are two steps to the remapping process. First w_k^y is calculated, then the result is multiplied by w_k^x to give the non-zero elements of $\mathbf{W}_{ij}$.

The calculation of each component of w_k^d is commutative and works identically for up to n dimensions. Our approach also ensures that $\sum_i W_{ij} \equiv 1$ which ensures that the number of particles in the simulation is conserved.



(a) *step 1* ***(b)*** *step 2*

Figure 3.5: *Calculating the redistribution of a cell over its neighbours consists of a series of processes equal in number to the number of dimensions. In two dimensions, the first step is to calculate the distribution of $\widehat{B}$ over the neighbours of $\widehat{A}$ in the vertical direction, based on the the translation of the cell centroid. In step 2, the translation of the centroid is used to make the same calculation in the horizontal. The results, w_k^x, and, w_k^y, are multiplied together to give the non-zero elements of one column of the discrete transfer operator $\mathbf{W}_{ij}$.*

3.2.1 Maximising algorithmic efficiency

We now briefly discuss how to achieve an efficient numerical implementation. In the proposed IRM method, trajectories drawn from the corner of each cell can be used for every cell sharing that corner. By sharing trajectories between multiple cells, the number of required particles in a cartesian grid is reduced to $(m+1)^d$, where m is the number of cells in each dimension, and d is the number of dimensions.

The discussed alternative of counting particles residing in each cell, requires the simulation of $N_p m^n$ particles, where N_p is the number of particles per cell. As $m \to \infty$, the computational benefit of the method utilising cells placed on corners, approaches the factor N_p. Both methods scale linearly with the number of grid cells and therefore have the same algorithmic complexity. The advantage of the IRM is that it can be conducted to higher orders of accuracy based on the complexity of the calculation of the intersecting volumes. Counting particles in cells will always be a first order method.

Memory usage is also an important factor. Consider a 2-dimensional space partitioned into N cells, arranged as in figure 3.6. The neighbourhood of the j_{th} cell is indicated by the light blue cells. When the discretised PDF of particle positions, P_j, is translated into a vector in row-major form, the neighbourhood of the j_{th} cell is not exclusively adjacent to the j_{th} cell in the vector. Where there is no link between the cells, the corresponding elements of $\mathbf{W}_{ij}$ are equal to zero. As the neighbourhood of each cell is limited to first order, the non-zero elements of $\mathbf{W}_{ij}$ are sparse, and appear in predictable diagonal stripes. When constructed in full, $\mathbf{W}_{ij}$ consists of N^2 elements, where N is the total number of cells in the discretisation of the particle phase space. However,

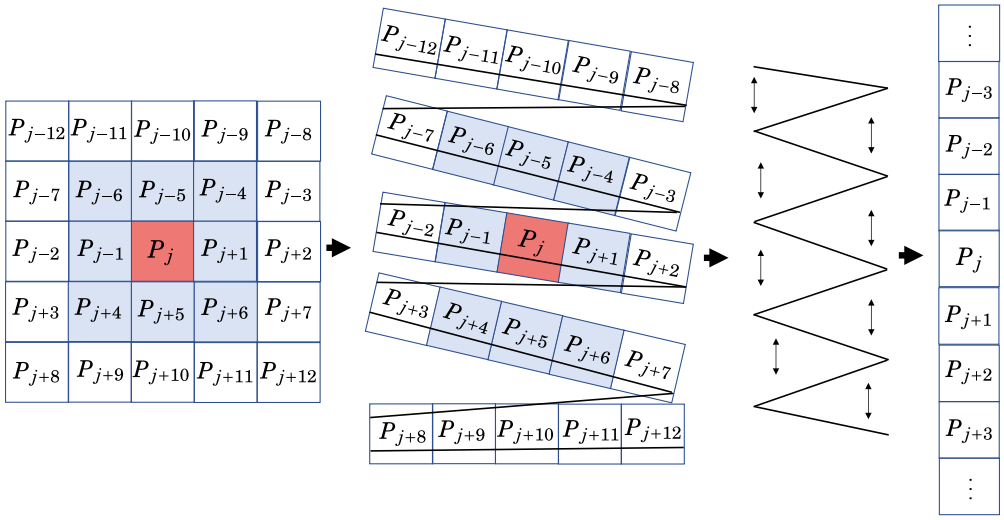


Figure 3.6: *The discretisation of the PDF of particle positions in phase space is given by P_j. Cells that are adjacent spatially, are not necessarily adjacent in the discrete vector, P_j. The figure shows the relationship between the spatial location of cells and their position in P_j. The positions are stretched out in a reverse concertina to produce P_j.*

when the matrix is stored for computational use, the diagonal strips of zeros need not be stored. By taking advantage of its sparsity, the required memory to store $\mathbf{W}_{ij}$ is reduced from N^2, to N times the number of neighbours of each cell. Each cell has 3^d first order neighbours, where d denotes the number of dimensions for which the transfer operator is constructed. A limiting factor of our implementation is that the cells can only be redistributed amongst their first order neighbourhood. Particles can move by no more than one cell per time step and so we must limit the time-step Δt. Having a constraint on the minimum time-step limits potential computational efficiency. However, when the method is combined with a numerical solver of the Navier-Stokes equations in the later chapters, the limitation on the step size is in line with limits on the solution of the fluid phase.

3.2.2 Boundary conditions

The PDF of particle positions in phase space is the solution of a partial differential equation (PDE). A PDE requires boundary conditions. We limit the dynamics of the system such that particles remain within some domain $\mathbf{G}$ for all time. That means that for $g(\mathbf{x})$, which is the numerical solution of the active tracer in equation 3.1, $\{\mathbf{x}, g(\mathbf{x})\} \in \mathbf{G} \forall t$. This condition can be enforced either by applying periodic boundaries, in which case $\mathbf{x} \in \mathbf{G}$ and $g(\mathbf{x}) = \mod(g(\mathbf{x}), \mathbf{G})$, or by enforcing that there is no flux through the boundaries of the domain.

For a box of size L, a no-flux condition requires $\dot{\mathbf{x}}(t, L) \cdot \mathbf{e}^{\perp} = \dot{\mathbf{x}}(t, 0) \cdot \mathbf{e}^{\perp} = 0$, where $\mathbf{e}^{\perp}$ denotes a unit vector perpendicular to the boundary. Whilst a no-flux condition would be imposed by the underlying fluid velocity field, inertial particles do not follow fluid streamlines. Under the

influence of a gravitational field, for example, aerosol particles will travel vertically relative to the fluid and pass through the bottom boundary of the domain. Therefore, our method must enforce its own boundary conditions.

Although boundary conditions could be imposed on the particles themselves to prevent them from leaving the domain, they are instead incorporated into the method through the use of ghost cells. Ghost cells form a frame around the domain into which fluxes are allowed to pass. After one time-step, and before the next, the concentration of particles that reside in the ghost cells are redistributed according to the type of boundary conditions that are proscribed by the specific problem. In a sedimentation problem, the bottom boundary will naturally be solid to simulate a river or sea bed. In a simulation of water droplets in a cloud, where only a small subsection of the cloud is simulated, periodic boundaries are applicable. Ghost cells are useful because they allow the same algorithm to be employed across the whole domain without requiring exceptions at the boundaries.

Two types of boundary condition are currently supported by computational implementations of the method: The first is a type of Neumann boundary condition we call a slip wall. When slip-walls are employed, fluxes parallel to the boundary behave as normal and fluxes through the boundary are set to 0. Slip walls are incorporated by adding the concentration of particles in the ghost cells to the next outermost column or row of cells. The diagram in figure 3.7 demonstrates the slip wall process. The outermost right column of cells is added to the next outermost column. Next, the bottom row is added to the penultimate bottom row. Any flux of particles through the boundary walls has been reversed, whilst fluxes along the boundary remain unchanged.

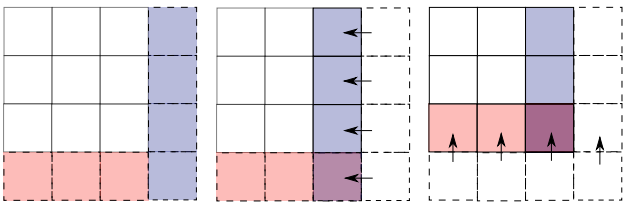


Figure 3.7: *One of the boundary conditions implemented in the transfer operator code is a 'slip wall'. Around the domain are a series of 'ghost cells' marked by the dashed lines. The concentration of particles in the ghost cells (outermost cells) are added to the next outermost cells to reverse any fluxes through the walls. In a sedimentation problem, the bottom row of cells fills with particles.*

The second type of boundary condition supported by our numerical implementation of the IRM is a periodic boundary condition. For periodic boundaries, the PDF of particle positions fluxed into the ghost cells is moved to the next outermost column or row on the opposite side of the grid, as shown in figure 3.8.

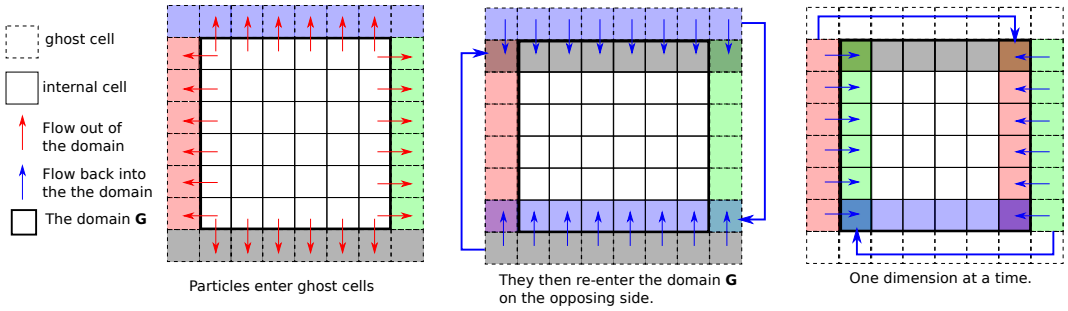


Figure 3.8: *The transfer operator code can be used with periodic boundary conditions. Particle concentrations in the outermost column are moved to the second outermost column on the opposite side of the grid, and particle concentrations in the bottom row are moved to the second from top row. This process is repeated for each dimension in turn.*

3.3 A comment on the nature of inertial particles

A transfer operator uses only the previous state of the system to determine the next state. The idea of a chain of states, the previous state, determining the next, is the definition of a Markov process. For this to work with inertial particles, the next state of the probability density field of particle positions must depend only on its previous state. In the case of inertial particles this is not necessarily an accurate assumption.

In general, the future position of a particle is dependent on its historical trajectory: an infinitely long memory of fluid and particle interactions that has a bearing on the particles current and future position and velocity. It would be theoretically possible to include the influence of history through use of a transfer operator Φ^t. By considering a limited history window that is discretised into a finite set of previous states of the system, as in the second chapter, the particle phase space including history becomes finite dimensional. However, producing a discretisation of Φ^t in a high dimensional phase space is numerically intractable. In the previous chapter, we demonstrated that there are instances where the history of a particle can be neglected. We limit ourselves to regions of the parameter space defined by S and ρ_p/ρ_f, where the Basset history force is insignificant.

Despite ensuring that the method is applied only in cases where the history force is insignificant a further complication remains. The future position of a particle is not only dependent on its current position in space, but also its current velocity. Figure 3.9 illustrates this last point. Three trajectories are simulated in the ABC flow with $\rho_p = 1000, \rho_f = 1$ and $S = 10^{-4}$. The black dashed line is an active tracer trajectory. The green line is a passive tracer started from the point X. The red line is a second active tracer trajectory. The second active tracer has been initialised at the velocity of the fluid at the point X. Clearly, the future trajectory of the particle is strongly dependent on its initial velocity: our transfer operator method must include particle velocity. More explicitly, each state in our Markov process must be $2n$ dimensional, where n is

the dimensionality of the particle configuration space.

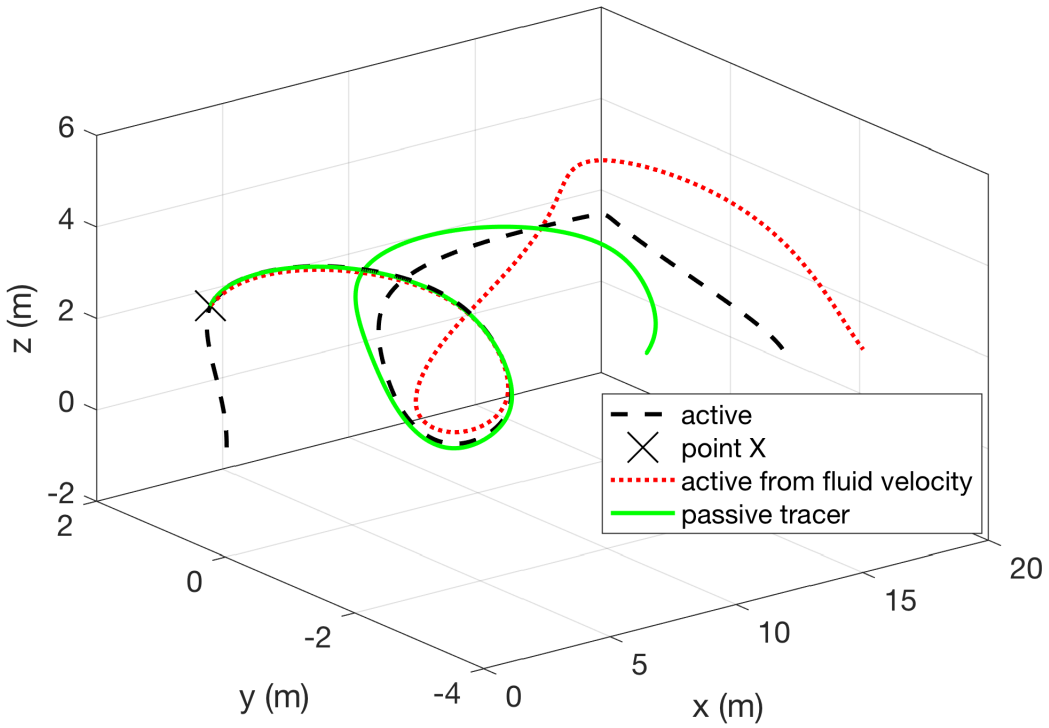


Figure 3.9: *The initial positions and velocities of inertial particles are vital to describe their future trajectories. Three trajectories have been simulated in the chaotic ABC flow with different initial conditions and modelling choices. First, an active tracer trajectory is calculated (black dashed line). Then, from a point intersected by the black dashed line (black cross), a passive tracer (green line) and a further active tracer (red dotted line), are initialised at the local fluid velocity. The three trajectories are markedly different. The particle parameters were given by* $S = 10^{-3}$*, and* $R \approx 10^{-3}$*.*

3.4 Method Verification

In the previous sections, we showed how a transfer operator for inertial particles could be constructed. In this section, we test our method in the simple 2-dimensional flow used in the second chapter, given by,

$$\mathbf{u}(\mathbf{x},t) = \begin{bmatrix} A\cos y + B\sin y \\ B\cos x + A\sin x \end{bmatrix}, \tag{3.11}$$

where A and B are constants governing the magnitude of the fluid velocity and the symmetry of the vortices. The flow consists of two periodic vortices and is therefore a good candidate to produce

particle clustering. Further, using a two dimensional flow is advantageous from the perspective of numerical complexity. As discussed previously, the position of a particle in configuration space is insufficient to uniquely determine its future trajectory. It is important that both the velocity and position of the particle are considered when constructing W_{ij}. A transfer operator constructed to act on particle position and velocity has twice the dimensions of the flow it seeks to approximate. When a flow, $\mathbf{u}(\mathbf{x},t):\mathbb{R}^2\rightarrow\mathbb{R}^2$, has a two dimensional configuration space, a 4-dimensional particle phase space is required to uniquely determine particle trajectories. Restricting ourselves to a 2-dimensional flow helps limit computational demands.

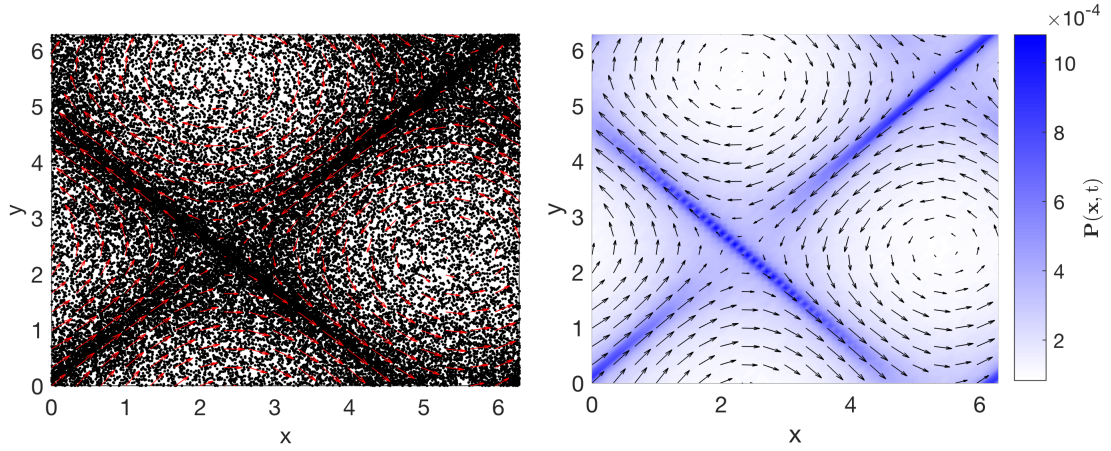
The four dimensional particle phase space associated with $\mathbf{u}(\mathbf{x},t)$ is defined and discretised as follows. The position of a particle in phase space is given by $\mathbf{x}=[x,y,u,v]$, where x and y are coordinates in configuration space, and u and v are coordinates in velocity space. Because the flow is 2π periodic, x and y are limited to $0<x<2\pi$, $0<y<2\pi$. The velocity space is limited so that $-\lambda U<u<\lambda U$ and $-\lambda U<v<\lambda U$, where λ is a factor that depends on the particle Stokes number $St=S/(RT)$, and U is the maximum domain wide fluid velocity. For large St, particle velocities deviate further from the fluid velocity, and the velocity dimensions must encompass a correspondingly wider domain. For all Stokes' numbers used in this study, $\lambda=1.3$ was empirically determined to sufficiently cover all possible particle velocities. Finally we introduce the discrete variables defining the discretisation. The particle phase space is discretised into $n_x\times n_y\times n_u\times n_v$ cells, where each of n_x, n_y, n_u and n_v denotes the number of cells in the x, y, u and v directions respectively. The grid resolution parameters, i.e the size of the domain divided by the number of cells in each respective dimension, are given by Δx, Δy, Δu and Δv.

To test our linearisation of a transfer operator for inertial particles, we produce a relative truth. The relative truth is a particle distribution generated through simulation of a large number of Lagrangian particle trajectories. Because the accuracy of our Lagrangian particle simulations was verified in the second chapter, it allows them to be used here as a benchmark for W_{ij}. We simulate 4×10^6 particles using equation 3.1, in the flow described by equation 3.11. The simulation is conducted for 0.5 characteristic timescales with flow parameters, A and B, set to 1 and 1.14 respectively. The particles are binned spatially over a cartesian grid consisting of 64×64 square cells to produce a distribution that can be compared with W_{ij}.

Distributions have been produced for a range of particle diameters and times, so that the accuracy of the transfer operator as a function of S/R can be assessed. To ensure that sufficient particles were simulated, we investigated how the distribution statistics changed with the number of particles. It was found that the skewness, mean, mode and standard deviations of the distributions had all converged for $N_p>10^6$. For comparisons with the discretisation of the transfer operator we use distributions generated from simulations with $N_p=4\times10^6$.

An example of the distribution resulting from the binning of particle positions is shown in figure 3.10 with particle size parameter equal to $S=1.07^{-4}$ and density parameter R equal to $R=\frac{2\rho_f}{2\rho_p+\rho_f}=10^{-3}$. The number of particles in each cell is normalised by the total number of

particles in the simulation to produce a PDF of particle positions. We denote the resulting discrete PDF of particle positions with $\widehat{\mathbf{X}}^i$, where i indicates the cell index.



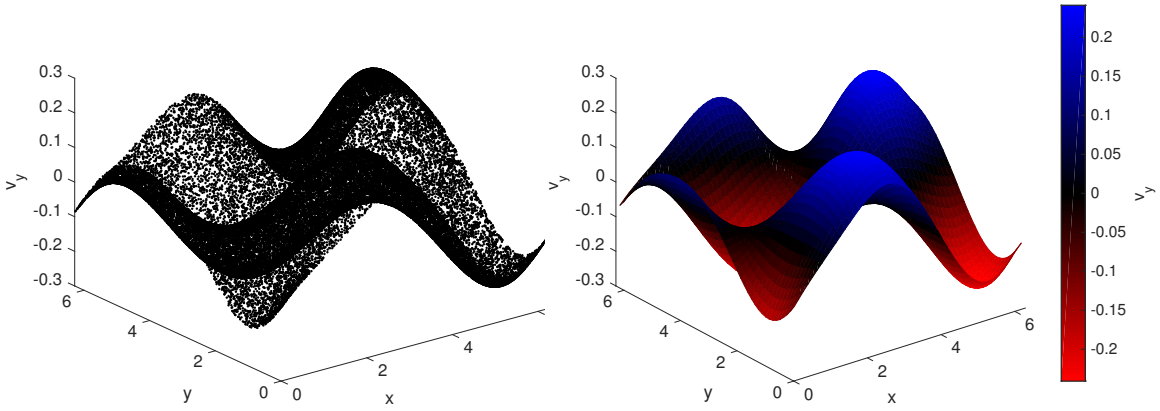
(a) *Before binning* ***(b)*** *After binning*

Figure 3.10: *Particle positions from Lagrangian particle simulations are binned to allow comparison with predictions made by the transfer operator. The figure shows the spatial distribution of particles before and after binning over a cartesian grid consisting of* 64×64 *cells. Over* 90% *of the particles have been excluded from (a) to improve clarity. In (b) the concentrations of particles have been normalised by the total sum of particles to produce a discrete PDF which can be compared directly with the transfer operator predictions.*

To investigate the velocity of particles across configuration space, a second particle distribution is produced. The second particle distribution consists of the average of the velocity difference $\mathbf{v}(t) - \mathbf{u}(\mathbf{x}, t)$ for each spatial grid cell. The velocities are attained by calculating the difference in velocity between each particle and the undisturbed fluid velocity in its location. The mean velocity difference is then calculated for each cell. Figure 3.11 shows an example of the velocity surface produced at the end of a 0.5 second simulation. The vertical positions of the particles are the y component of the velocity difference $\mathbf{v}(t) - \mathbf{u}(\mathbf{x}, t)$. The particle size parameter was $S = 1.07^{-4}$ and the density ratio was set to $R = \frac{2\rho_f}{2\rho_p + \rho_f} = 10^{-3}$. Figure 3.11 (a) shows the difference between the local fluid velocity and the velocity of each particle before binning; figure 3.11 (b) shows the surface constructed from the average velocity difference in each bin. The cell wise particle velocity used to produce the surface in figure 3.11, will be denoted by $\mathbf{V}_i$, where i is the cell index.

In the following, the two reference distributions, $\mathbf{V}_i$ and $\widehat{\mathbf{X}}_i$ will be used to investigate the accuracy of the IRM. To summarise, we have simulated a large number of active tracer particles in a steady 2-dimensional flow described by equation 3.11. The particles are assigned to discrete cells in configuration space to produce a discrete particle probability distribution given by $\widehat{\mathbf{X}}_i$. The velocity of the particles in each bin, which produces a velocity surface over configuration space, is denoted by $\mathbf{V}_i$. The undisturbed fluid velocity is denoted by $\mathbf{u}(\mathbf{x}, t)$.

A further point of useful comparison is the inertial equation derived in [4], and presented in



(a) *Before binning* ***(b)*** *After binning*

Figure 3.11: *For small enough S/R, particle velocities are predictable and it is possible to produce a velocity surface over configuration space. (a) The y component of the particle velocity, v_y, minus the y component of the fluid velocity, v across the $2\pi \times 2\pi$ space reveals a smooth surface. (b) The smooth surface can be binned spatially over a 64×64 cartesian grid by finding the average particle velocity in each grid square. A surface of the average value of $v_y - v$ is useful for comparison with the transfer operator simulations.*

equation 2.13. Haller and Sapsis, motivated by problems of source inversion for inertial particles, derived an equation, called the inertial equation, that describes the velocity of an inertial particle given the local fluid velocity. The inertial equation represents a manifold in phase space to which particle trajectories are attracted. A particle released from some position $\mathbf{x}(0)$ and velocity $\mathbf{v}(0)$, will be attracted exponentially towards a trajectory on the manifold. Figure 3.12 gives a visual representation of the concept. A family of initial conditions, linked by a stable fibre, converge to a particle trajectory on the manifold. The surface in figure 3.12 is known as an inertial manifold which is a perturbation of the underlying fluid velocity. The size of the perturbation is determined by the small parameter $\epsilon = S/R$. For very small particles, the manifold is simply given by the undisturbed fluid velocity. For larger particles, the shape of the manifold becomes less similar to undisturbed fluid velocity. Parallels between the surface the inertial manifold and the velocity surface in figure 3.11 should be immediately obvious. In fact, for modest S/R the two surfaces are equivalent. The velocities of particles described by the inertial equation will be denoted by $\mathbf{m}(x, y)$.

Each of the measures above, the binned Lagrangian particle velocities, $\mathbf{V}_i(t)$, the undisturbed fluid velocity $\mathbf{u}(\mathbf{x}, t)$ and the velocity field predicted by equation 2.13, will be used to assess the accuracy of P_j^n.

First we assess the ability of the IRM to predict particle velocities over configuration space. Consider the planar slice through the velocity dimensions shown in figure 3.13. The slice through the domain is given by $[x = \pi, y = \pi, u, v]$. The undisturbed fluid velocity, $\mathbf{u}([\pi, \pi], t)$ is indicated

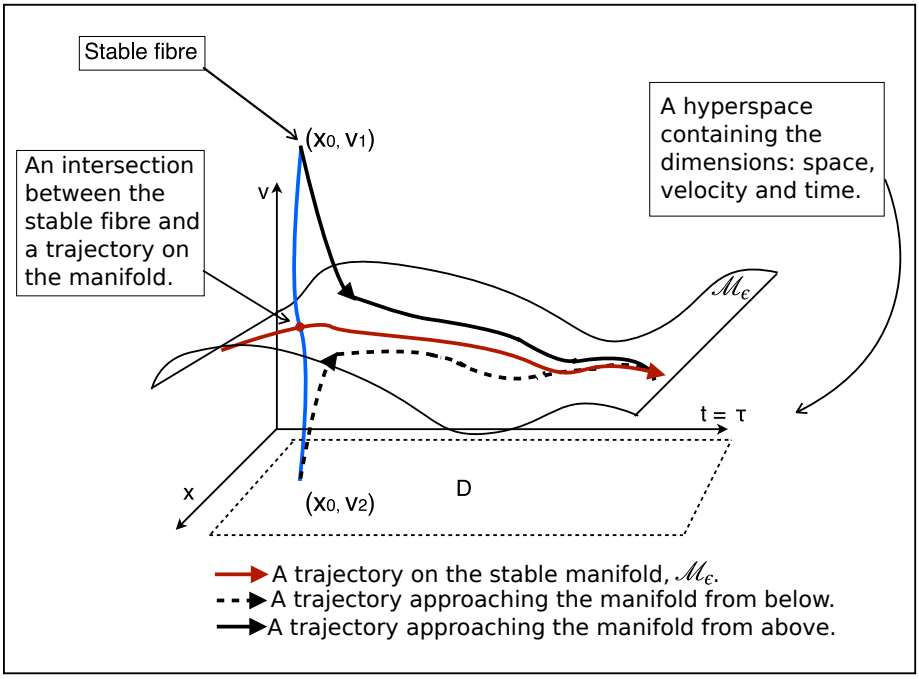


Figure 3.12: *The velocity surface in figure 3.10 is an example of a slow stable manifold. Two inertial particle trajectories are shown approaching a manifold, $\mathcal{M}_\epsilon$, forwards in time. The particle trajectories converge exponentially to a trajectory on $\mathcal{M}_\epsilon$; a stable fibre, intersecting $\mathcal{M}_\epsilon$, represents a family of initial conditions that converge to the red trajectory on the manifold. The diagram is inspired by a similar diagram in Haller and Sapsis [4].*

by the red point, and the average particle velocity from the particle simulation, $\mathbf{V}([\pi,\pi],t)$, is marked by the black point. The velocities of each individual Lagrangian particle are also shown, marked by the smaller grey squares, and the velocity on the manifold, $\mathbf{m}(x,y)$, is indicated by the green cross. Finally, $P([\pi,\pi],u,v)$ is represented by the colour map. In figure 3.13, $P([\pi,\pi],u,v)$ has a maximum at the same particle velocity as predicted by the particle simulation and the inertial equation, indicating both that W_{ij} is working, and that the inertial equation is a good representation of particle velocities for this value of S/R.

For every point in configuration space, a similar figure can be produced. From each similar slice of $\mathbf{P}_j$, a single velocity coordinate can also be obtained. We use two methods to derive a single velocity coordinate for each slice: the first is the velocity position of the maxima of $\mathbf{P}_j$ as a function of configuration space, given by $\mathbf{Q}_i(\mathbf{x},t) = \arg\max_{\mathbf{v}}(\mathbf{P}(\mathbf{x},\mathbf{v},t))$; the second is the centroid of $\mathbf{P}_j$, with respect to the velocity dimensions, given by $\widehat{\mathbf{Q}}_i(\mathbf{x},t) = \frac{\sum_k \mathbf{v}_k \mathbf{P}_{ik}}{\sum_k \mathbf{P}_{ik}}$. Here we allow $\mathbf{P}_{ik}$ to denote the probability density at the position $[\mathbf{x}_i,\mathbf{v}_k]$, where $i = 1,2,3...n_x$ and $k = 1,2,3...n_u$.

Errors on the velocity will be given by the Euclidian distance between the transfer operator measures, $\mathbf{Q}_i$ and $\widehat{\mathbf{Q}}_i$, and the particle velocity distribution $\mathbf{V}_i$, predicted by the particle simulation.

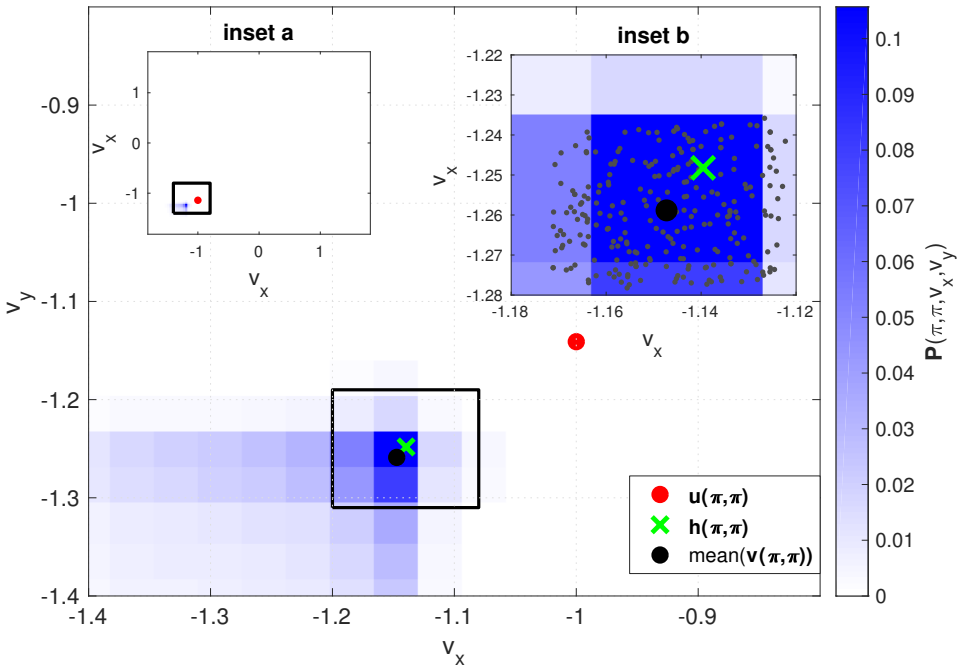
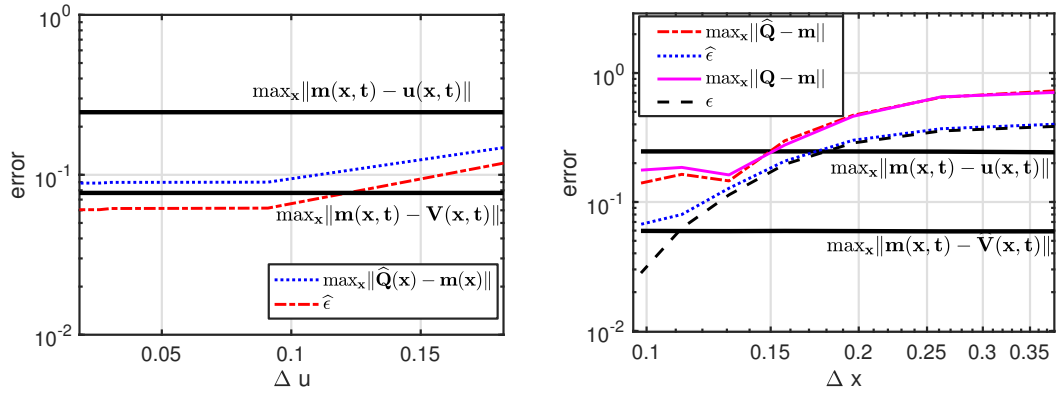


Figure 3.13: *We can investigate how well the transfer operator predicts particle velocities by plotting a velocity plane for a point in configuration space, $[x = \pi, y = \pi]$. The discretised PDF, $P(\mathbf{x}, \mathbf{v}, t)$ is at a maximum close to the average Lagrangian particle velocity (black point). The velocity on the manifold (green cross), predicted by equation 2.13, is also a good approximation of the Lagrangian particle velocity, whereas the undisturbed fluid velocity is a poor approximation (red point). The number of grid cells in each dimension is given by $n_x = n_y = 20$, $n_{vx} = n_{vy} = 100$.* ***Inset a:*** *shows the position of the velocities with respect to the whole velocity space.* ***Inset b:*** *shows an enlarged view of the maximum of $P([\pi, \pi], \mathbf{v}, t)$. The distribution of Lagrangian particle velocities around their mean is plotted as smaller grey points.*

The error is a function of configuration space, and varies depending on the local undisturbed fluid acceleration. We define the local errors as $\delta_i = \|\mathbf{Q}_i - \mathbf{V}_i\|$ and $\widehat{\delta}_i = \|\widehat{\mathbf{Q}}_i - \mathbf{V}_i\|$, where $\|\cdot\|$ denotes the Euclidian norm. The global errors we define as the spatial maximum of δ_i and $\widehat{\delta}_i$, and denote these by $\epsilon_i = \max_i(\delta_i)$ and $\widehat{\epsilon} = \max_i(\widehat{\delta}_i)$ respectively. The errors here are not normalised as the values of the velocities are all $O(1)$.

Figure 3.14 shows how ϵ and $\widehat{\epsilon}$ vary with grid resolution. Both the spatial resolution of the grid Δx and the resolution of the grid in the velocity dimensions Δu have been investigated. The thick black lines in each of figures 3.14(a) and (b) are plotted as useful reference points. One shows the maximum difference between the undisturbed fluid velocity and the velocity predicted by the manifold, $\max_{\mathbf{x}}(\|\mathbf{u}(\mathbf{x}, t) - \mathbf{m}(\mathbf{x}, t)\|)$. The other shows the maximum difference between $\mathbf{V}_i$ and the velocity on the manifold, $\max_{\mathbf{x}}(\|\mathbf{V}_i - \mathbf{m}(\mathbf{x}, t)\|)$. The error of the transfer operator has been plotted as a function of both Δx and Δu. A minimum in the error is reached for figure 3.14

(a). For greater accuracy, we require further increases in spatial resolution, but are limited by the memory requirements of the method.



(a) *reducing* Δu ***(b)*** *reducing* Δx

Figure 3.14: *We are interested in how the difference between the transfer operator and the Lagrangian particle simulations changes as a function of the two numerical parameters: the cell size in configuration space* Δx*, and the cell size in velocity space,* Δu*. (a) The total error on the particle velocity falls for increasing resolution in the velocity dimensions, up to* $\Delta u = 0.1$*. (b) The total error on the particle velocity predicted by the transfer operator continues to fall for increasing spatial resolution.*

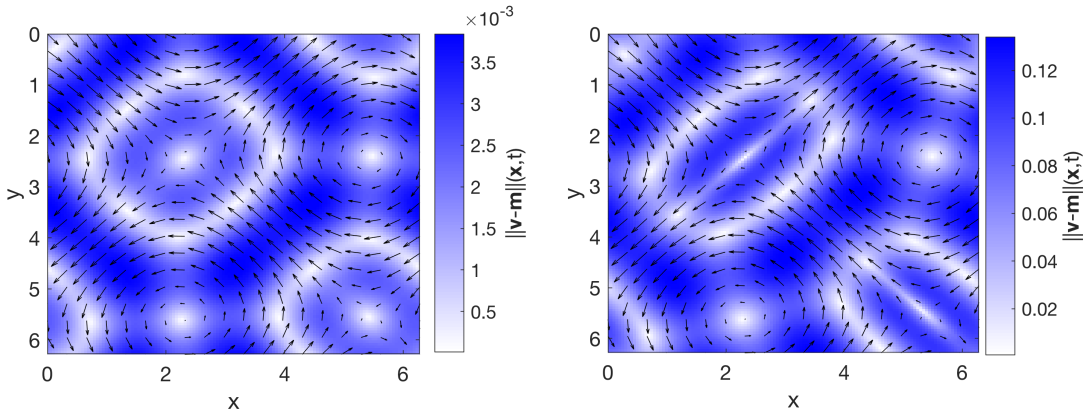
3.4.1 Reducing the dimensions

Whilst discretising phase space is feasible when the configuration space is only 2-dimensional, (4-dimensional transfer operator), the tractability to a 3-dimensional configuration space (6 dimensional transfer operator) is limited. Discretising a 6-dimensional space results in large numbers of cells. We take the example of blue crystal, the University of Bristols high powered computing resource. Each processor on phase 3 of blue crystal has a memory capacity of $4GB$. In the most efficient implementation of the transfer operator, we require a minimum of 2 doubles per grid cell to store the new and old state of the grid. If each dimension contains the same number of cells, m, the total number of cells in the discretisation is m^6. A double requires 8 bytes and so, in total, we require $16 \times m^6$ *bytes*. Therefore, the maximum number of cells in each dimension on an implementation on one processor on blue crystal is $m = \left(\frac{4\times 10^9}{16}\right)^{\frac{1}{6}} = 25$. Even for a four dimensional phase space, the achievable resolution is limited. As demonstrated in the previous section we require more than 64 cells in the velocity dimensions to achieve greater velocity accuracy than provided by the inertial equation. Clearly this is not the computational efficiency we sought. Fortunately there is a better solution.

Within the particle phase space there are regions that are more likely to be occupied by particles than others. In fact, for weakly inertial particles, the particle velocity can be uniquely defined everywhere in space as a subtle deviation from the fluid velocity. The inertial equation,

derived in [4], and given in equation 2.13, defines the particle velocity deviation from the fluid velocity everywhere in configuration space. By assuming that the particle dynamics lie close to the invariant manifold described by the inertial equation, the dynamics of particles can be assumed to be Markovian; the velocity dimensions can be collapsed. For small Stokes number problems, such as the early stages of raindrop formation, the “inertial equation" is a globally attracting surface [86]. For some applications then, where particles lie within a specific parameter regime, the inertial equation is sufficient to capture the inertial behaviour of the particles.

The validity of a dimensionality reduction is dependent on the Stokes number of the particles. For small particle diameters, the particle velocities form a surface over the configuration space as in figure 3.11: a surface that closely matches the invariant manifold described by the inertial equation. Figure 3.16 shows the surface $\|\mathbf{v}-\mathbf{m}\|(\mathbf{x},t)$ for two values of S/R, 0.027 and 0.671. For $S/R = 0.027$ the surfaces are closely matched and errors are of the order of the grid resolution, 10^{-3}. For $S/R = 0.671$ some regions produce large differences between the two surfaces. For parameter regimes where the two surfaces are closely matched, the inertial equation can be exploited to allow the future trajectory of an inertial particle to be determined without knowledge of its past velocity. With unique knowledge of a particle's velocity, a transfer operator can be constructed that acts only on the particle configuration space.



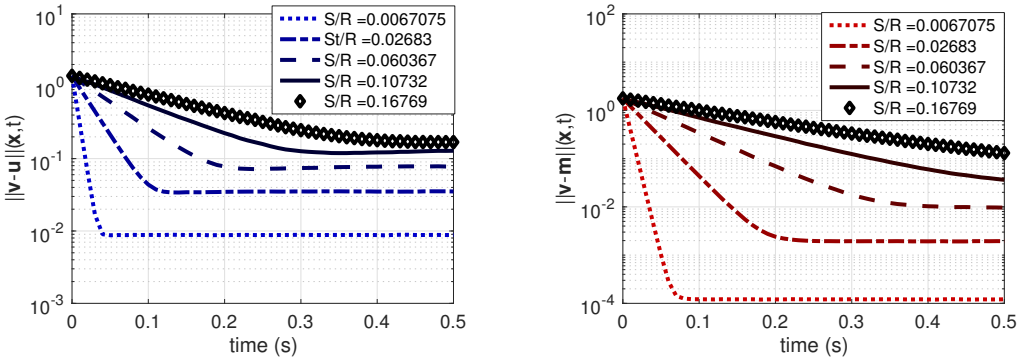
(a) $S/R = 0.027$ ***(b)*** $S/R = 0.671$

Figure 3.15: *The Euclidian distance between Lagrangian particle velocities and the velocity calculated using equation 2.13, represented as a colour map, is a function of S/R. Values are calculated by binning Lagrangian particles in discrete cells and calculating the average velocity in each cell. The accuracy of the inertial equation is expected to have a strong dependance on S/R, as equation 2.13 was derived using a power series expansion around $S/R = 0$ [4].*

In this section we assess how the accuracy of a transfer operator, acting only on the PDF of particle positions in configuration space, changes with increasing values of the parameter S/R. As before, the space is discretised into a regular cartesian grid and particles are initialised on the corners of the grid cells. However, rather than discretising the velocity dimensions, the initial velocity is calculated using the “inertial equation". In some instances reducing the dimensions

will prove successful, in others the assumption that the particles lie close to the manifold breaks down.

Figure 3.16 shows how the velocities of particles in a particle simulation are attracted to the manifold for a range of size parameters. The particle initial conditions are randomly sampled from a uniform distribution over the range $0 < x(0) < 2\pi$, $0 < y(0) < 2\pi$, $-\lambda U < u(0) < \lambda U$ and $-\lambda U < v(0) < \lambda U$. Taking a wide range of initial velocities allows us to assess the decay of the velocity difference $\|\mathbf{v}-\mathbf{m}\|(\mathbf{x},t)$ as a function of space and time. Figure 3.16 (b) shows the decay of $\|\mathbf{v}-\mathbf{m}\|(\mathbf{x},t)$ averaged over the configuration space. For small S/R the attraction to the manifold is rapid. For $S/R < 0.1$ the velocity difference $\|\mathbf{v}-\mathbf{m}\|(\mathbf{x},t)$ reaches a steady state within 0.12 characteristic timescales. For larger S/R the attraction is slower. For $S/R = 0.10732$ and 0.16769 the decay is too slow to reach a minimum. As expected, the velocity on the manifold is further from the actual particle velocity for larger S/R.



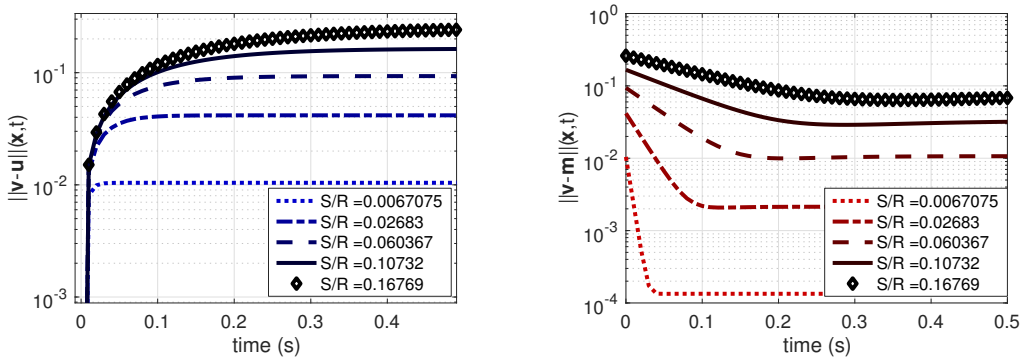
(a) *average difference between the particle and fluid velocity.*

(b) *average difference between the particle velocity and the manifold.*

Figure 3.16: *The Euclidian distance between Lagrangian particle velocities and the undisturbed fluid velocity (a), and velocities predicted by equation 2.13 (b) are plotted as functions of time. Velocity differences decay more slowly as particle diameters increase. For large particles, $S/R > 0.1$ the manifold is a poor prediction of the particle velocity. Initial particle velocities are taken from a uniform distribution over the interval $[-U - 0.3 < v_0 < U + 0.3]$.*

A uniform distribution of initial particle conditions across phase space is not a realistic assumption: in reality, particle velocities are likely to be close to the fluid velocity. By initialising the particles at the local fluid velocity we can reach the average, domain wide, steady state velocity error within 0.12 characteristic timescales for the full range of particle parameters. Figure 3.17 demonstrates this result.

The inertial equation is a good prediction of the local particle velocity. We are now able to justify the dimensionality reduction for $S/R \lesssim 0.1$. In the following, we investigate the accuracy of our transfer operator approach for modelling particle clustering.



(a) *average difference between the particle and fluid velocity.*

(b) *average difference between the particle velocity and the manifold.*

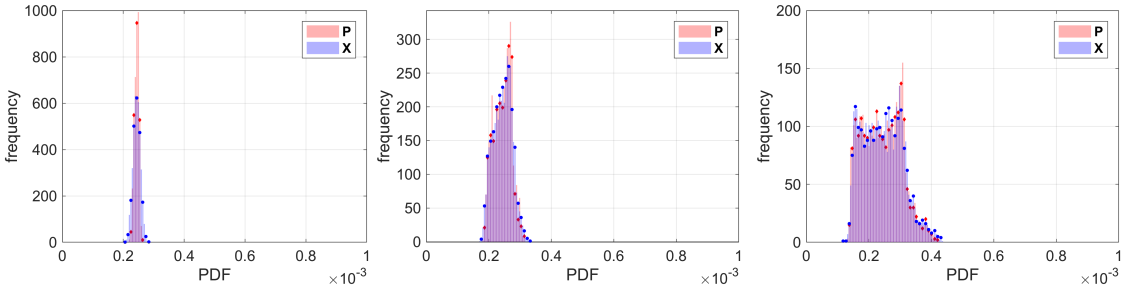
Figure 3.17: *In figure 3.16, the decay of velocity differences between Lagrangian particles and velocities predicted by equation 2.13 (b), did not reach their minimum. To remedy that, particle velocities are initialised at the at the local fluid velocity to produce this figure. Particle velocities diverge from the local fluid velocity; for all S/R the inertial equation is a considerably better prediction of particle velocities.*

3.4.2 Results

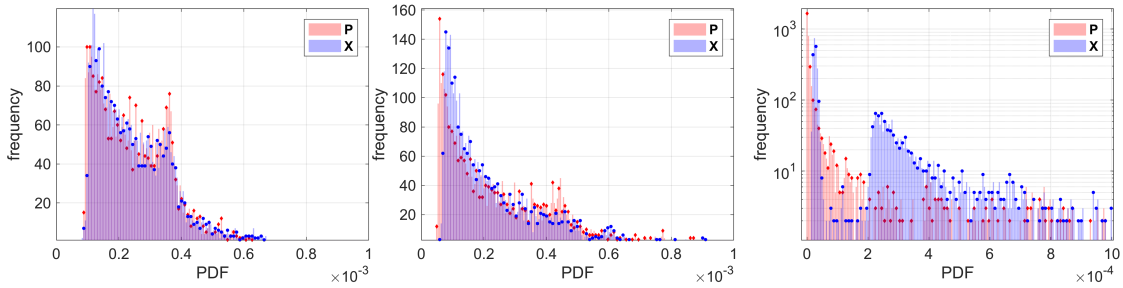
Our motivation for a transfer operator approach is to be able to describe the particle concentration everywhere in space in a computationally efficient manner. Given an understanding of the accuracy of the inertial equation at predicting the particle velocity, we now investigate the degree of clustering predicted by the transfer operator in the simple flow described by equation 3.11. As a comparison we use the degree of clustering predicted by the particle simulation for a range of values of S/R. Particle clustering is a measure of the degree to which particles group together in certain regions of the flow.

A good measure of particle clustering is to compare frequency distributions created from P_j and X_j. To create frequency distributions, P_j and X_j, are sorted in ascending order, and binned, with a count associated with each bin. The degree of clustering displayed in each distribution is encapsulated in the distribution skewness and mode. The further skewed to the left the frequency distribution, the greater the degree of clustering. When clustering is pronounced, more cells exist with few particles and so the peaks of the frequency distributions move to the left. Similarly, there are some rare cells that contain very many particles which produces long tails in the positive direction. Figure 3.18 shows how the clustering shown in P_j^n and X_j vary with the parameter S/R. The distributions match well for small diameters, but become less accurate as the parameter S/R increases. The degree of clustering is well maintained for $S/R \leq 0.168$.

By extracting characteristics of the frequency distributions in figure 3.18, we can compare the clustering predicted by W_{ij}, with the clustering from the Lagrangian particle simulations, in a more quantitative manner. Figure 3.19 shows how the modes, skewness and standard deviations of the frequency distributions change as a function of S/R. At approximately $S/R = 0.1$,



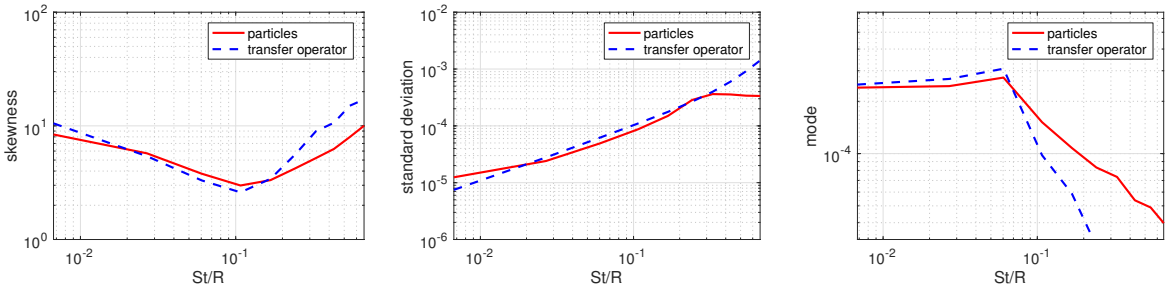
(a) $S/R = 0.007$ ***(b)*** $S/R = 0.027$ ***(c)*** $S/R = 0.06$



(d) $S/R = 0.107$ ***(e)*** $S/R = 0.168$ ***(f)*** $S/R = 0.671$

Figure 3.18: *A comparison between frequency distributions of* $\mathbf{P}(\mathbf{x},t)$ *and* $\mathbf{X}(\mathbf{x},t)$ *allow us to investigate the success of* W_{ij} *as a function of* S/R*. The distributions were created after* 0.5 *characteristic timescales in the flow described by equation 3.11 with* $A = 1$ *and* $B = 1.14$*. The number of cells in* $P(\mathbf{x},t)$ *was* $n_x = n_y = 1024$ *which has been resampled to match* $X(\mathbf{x},t)$*, where* $n_x = n_y = 64$*. The two match well visually up to approximately* $S/R = 0.1$*. For* $S/R = 0.671$ *the discrepancies are marked.*

discrepancies begin to appear between distribution measures of P_i and X_i, indicating that the accuracy of W_{ij} is limited to approximately $S/R < 0.1$.



(a) *skewness* ***(b)*** *standard deviations* ***(c)*** *mode*

Figure 3.19: *Extracting the skewness, modes, and standard deviations of the distributions in figure 3.18 give us a clearer picture of the accuracy of* W_{ij}*. The distributions show good agreement up to a value of* $S/R \approx 0.1$*. For* $S/R > 0.1$ *the behaviour of the particles begins to change.*

For very small S/R, where there is almost no vortex ejection, the modal bin in figure 3.18(a) is more heavily filled by W_{ij} than by the Lagrangian particle simulation. The cell concentration,

$Np/cell$, in the particle simulation is closer to a gaussian distribution than the cell concentration predicted by W_{ij}. Hint of a gaussian distribution is evidence that the Lagrangian particle spatial statistics display randomness. If we are to compare local values of the PDF produced by the transfer operator with the PDF from the Lagrangian particle simulation, as one would with error analysis of a PDE, we desire greater spatial smoothness of Lagrangian particle positions. To reduce issues of noise from the particle simulation, we investigate the efficacy of a simple, centre weighted, moving average smoothing. The scheme is given by,

$$\widehat{\mathbf{N}}_{\mathbf{p}}^{(\mathbf{i},\mathbf{j})} = \frac{2N_p^{(i,j)} + N_p^{(i-1,j)} + N_p^{(i,j-1)} + N_p^{(i+1,j)} + N_p^{(i,j+1)}}{6} \tag{3.12}$$

where $N_p^{(i,j)}$ denotes the number of particles in cell (i,j) of the discretisation and i and j denote the row and column index respectively. The values in $N_p^{(i,j)}$ are the same as those as in $\mathbf{X}_i$ although $N_p^{(i,j)}$ has been transformed to a matrix so that the positions of its elements are aligned with their corresponding position in space. We use one iteration of the smoothing algorithm in equation 3.12, to smooth the particle positions in space. Further iterations of smoothing serve only to smear features in the particle distribution. The sharpness of clustering along the separatrix is reduced if too many iterations of smoothing are conducted.

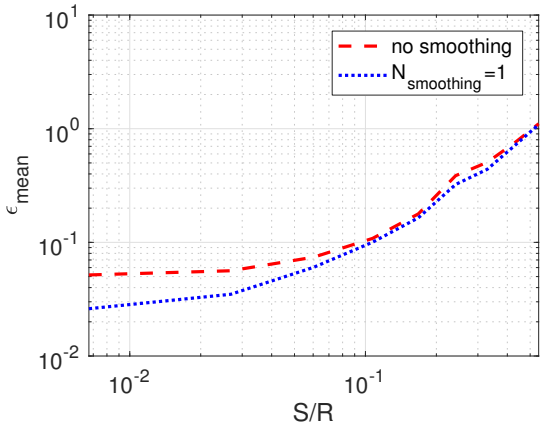


Figure 3.20: *The accuracy of predictions made with W_{ij} is a strong function of S/R. The figure shows the difference between the Lagrangian particle distribution, $\mathbf{X}_j$, and the distribution predicted by W_{ij}, P_j. At $S/R \approx 0.1$ smoothing no longer serves to reduce the error implying that errors are no longer related to randomness in the particle distribution. At this point, the linearisation of the transfer operator begins to misrepresent particle dynamics.*

With a smoother particle distribution, we now compare the local values of P_j with X_j. Our error measure is given by,

$$\epsilon_{mean} = \frac{\sum_j |X_j - P_j|}{\sum_j P_j}, \quad j = 1,2,3...N, \tag{3.13}$$

where we have normalised the absolute difference between the PDFs, with the average of P_j. Figure 3.20 shows that the average difference between the two PDFs is a strong function of S/R. In fact, at $S/R \approx 0.1$ we see again a clear change in behaviour. For $S/R > 0.1$ smoothing of the particle distribution has no influence on the accuracy of our method, which implies that at this point, the particle behaviour begins to change. We will investigate exactly how the particle behaviour changes in the later sections.

3.5 Particle behaviour for large S/R

In the previous section, we saw that at approximately $S/R = 0.1$, the accuracy of W_{ij} starts to reduce dramatically. We want to understand exactly what it is about the particle dynamics that cause the accuracy of our method to be reduced.

In our method, we use Haller's inertial equation to calculate initial particle velocities. We are therefore reliant on the inertial manifold in figure 3.12 remaining an attracting surface for all space and time. In general, the manifold is not always an attracting surface. Conditions on the stability of the inertial manifold were derived in [4] for the case of neutrally buoyant particles ($R = 2/3$). Conditions of stability are based on the rate of strain tensor $\mathbf{E} = \nabla\mathbf{u}(\mathbf{x},t) + \nabla\mathbf{u}(\mathbf{x},t)^T$, where superscript ^{T} denotes the matrix transpose. The rate of strain tensor is used to construct the matrix $\mathbf{M} = \left[E(\mathbf{x},t) + \frac{2}{3S}I\right]$, where $\mathbf{I}$ is the identity matrix. The inertial manifold will locally repel nearby trajectories if,

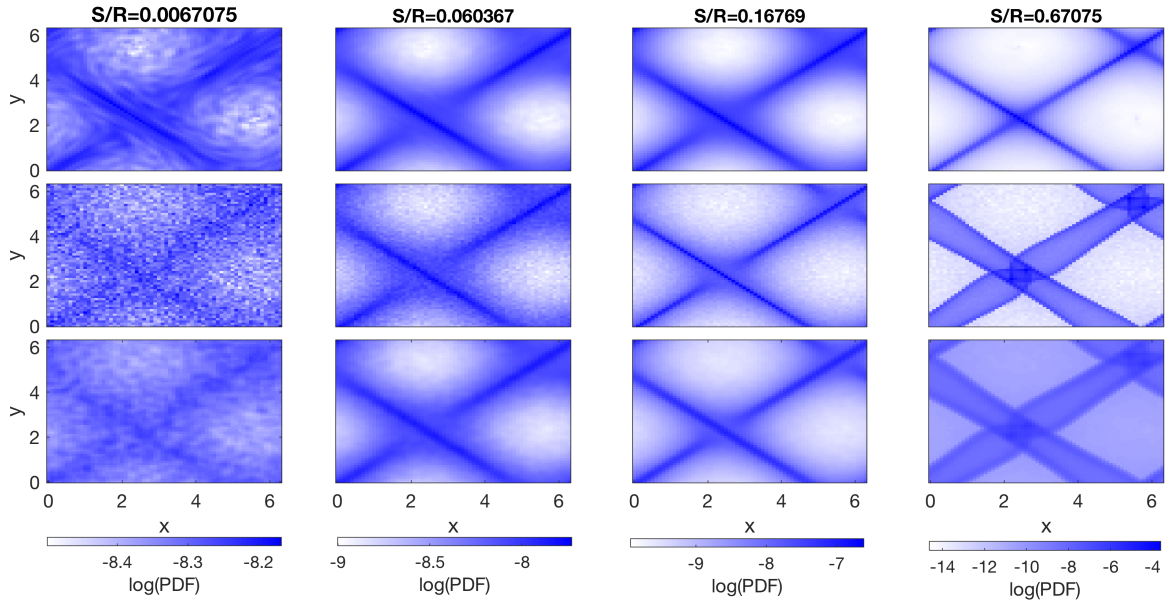
$$\lambda_{min}(\mathbf{M}) < 0, \tag{3.14}$$

where $\lambda_{min}(\mathbf{M})$ represents the smallest eigenvalue of $\mathbf{M}$. Instabilities in the inertial manifold occur either when fluid accelerations are large, or S/R is large. For small values of S/R, particle trajectories lie close to the inertial manifold, but as S/R increases in value, the particle velocity deviates from the manifold. The question is, what does that mean for the particle dynamics?

Figure 3.21 shows the probability density function of particle positions for four values of S/R. The panels in the vertical show the PDF predicted by W_{ij} operator (top), the binned particles (middle row), and the binned particles after one iteration of smoothing (bottom row). For $S/R = 0.0067$, $S/R = 0.0604$ the behaviour of the particles is well represented by W_{ij}. For $S/R = 0.167$, whilst qualitatively similar, the rate of vortex ejection is underestimated by the transfer operator, a behaviour that can be attributed to numerical diffusion. However, when the particle diameter is increased further still, the orbits taken by the particles fundamentally change. For the case of $S/R = 0.6707$ the Lagrangian particles are attracted to a range of orbits. Particles travel along a number of different paths, close to, but not along the separatrix. The transfer operator on the other hand predicts that particles will be attracted to the separatrix, so that they are concentrated along it.

The behaviour displayed in figure 3.21(d) explains why the errors in figure 3.20 are so large for larger values of S/R. The maximum domain wide error occurs in regions for which the transfer

operator predicts there to be no particles, but in reality the particles are travelling down one of the alternative orbits close to, but not along, the separatrix.



(a) $S/R = 0.0067$ ***(b)*** $S/R = 0.0604$ ***(c)*** $S/R = 0.1677$ ***(d)*** $S/R = 0.671$

Figure 3.21: *A visualisation of the PDFs of particle positions provide a physical understanding of why the accuracy of predictions made with $W_{i,j}$ decrease with increasing S/R. The distribution of particles is shown after* 0.5 *characteristic timescales for a range of size parameters. Particles with large size parameters do not line up along the separatrix, but instead are attracted to a band of orbits. The transfer operator is unable to capture this behaviour.*

3.6 Grid convergence study

In this section we conduct a grid convergence study comparing a series of lower resolution simulations to a high resolution benchmark. The convergence study has been carried out for steady flows in both two and three dimensional configuration spaces. For the 2-dimensional case, we use the flow given by equation 3.11. For the three dimensional test flow we use the ABC flow given by,

$$\mathbf{u}(x,0) = \begin{bmatrix} C\cos(y) + A\sin(z) \\ A\cos(z) + B\sin(x) \\ B\cos(x) + C\sin(y) \end{bmatrix}, \tag{3.15}$$

where A, B and C are flow constants that control the velocity of the flow. In this case we set $A = 1$, $B = \sqrt{2}$, and $C = \pi$. For comparison between simulations of differing resolution, values in the cells from the high resolution run are summed to match the number of grid cells of the lower resolution runs.

Let the hatted variable $\widehat{P}_j$, represent the discrete PDF of particle positions from the highest resolution simulation. The PDF for a lower simulation run is represented by P_j. The error measures are defined as the maximum, $\max_j[|\widehat{P}_j - P_j|]$, and average, $\text{mean}[|\widehat{P}_j - P_j|]$, domain wide difference between the high resolution simulation and a lower resolution simulation. For the benchmark runs we use simulations consisting of 2048^2 and 256^3 in two and three dimensions respectively.

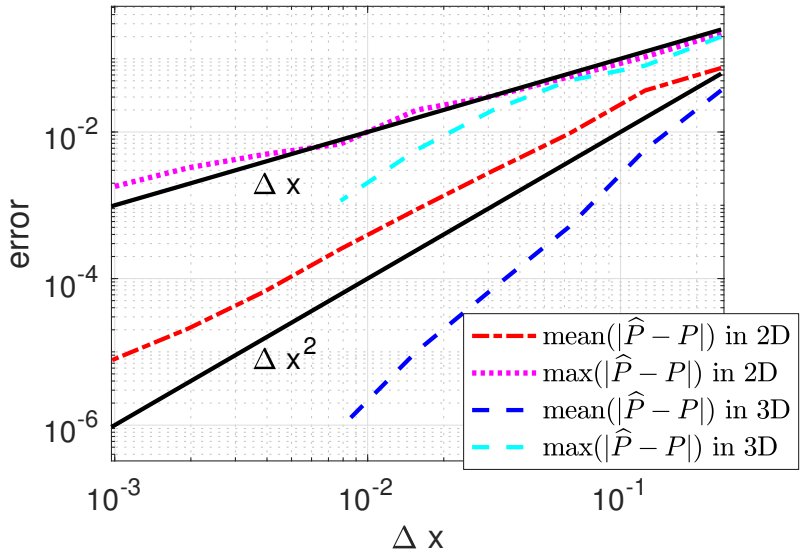


Figure 3.22: *Method convergence is investigated through a comparison between the PDF of particle positions from a high resolution simulation, and a series of lower resolution simulations. Tests were conducted both in the three dimensional flow given in equation 3.15, and the two dimensional flow given in equation 3.11. In two dimensions, the high resolution run consisted of* 2048^2 *cells, and in three dimensions the domain was divided into* 256^3 *cells. The particle size parameter was* 3.5×10^{-4}*, the density parameter was* $R = 10^{-3}$ *and the flow constants were given by* $A = 1$*,* $B = \sqrt{2}$ *and* $C = \pi$*.*

Figure 3.22 shows how the error scales for increasing spatial resolution.Interestingly, the error converges more quickly in three dimensions than in two. In three dimensions, the method convergence is between 1st and 2nd order, whereas in two dimensions, the method is clearly 1st order. In 2-dimensions the flow includes a saddle point; the maximum error in the domain always appear over it. Consider the saddle point in figure 3.23. The centroid of a cell with vertices that encompass the saddle will remain fixed in its original location. Because the centroid remains fixed, mass congregates in the cell and is not transported along the separatrix. The ABC flow has no such saddle point and as such the cells perform homogeneously throughout the domain.

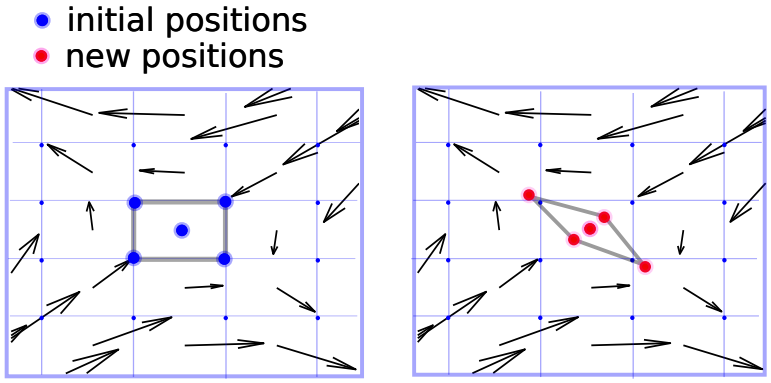


Figure 3.23: *When a cell encompasses a saddle point, the first order remapping method fails to capture the redistribution of mass along the stable manifold. Despite particles moving, the symmetry of their movement means that the cell centroid remains in place. Higher order remapping processes would remove this source of error.*

3.7 Eigenfunctions of the transfer operator

An attractive property of a transfer operator is that the long term behaviour of the system is captured in the eigenfunctions of the operator. The long term probability distribution of the particle positions can be found in the eigenvector corresponding to the largest eigenvalue of W_{ij}. For large matrices, power iteration of the linear system, $P^n = W_{ij}^n P^0$, provides an efficient approach for finding the dominant eigenvector of W_{ij}. In a steady flow, repeated multiplication of the vector of particle cell densities by the transfer operator provides an efficient method for determining the long term clustering regions of the flow.

Reducing the dimensions of the system such that only configuration space is discretised, allows us to construct W_{ij} in 3-dimensional flows. Figure 3.24 shows the dominant eigenvector of the transfer operator applied to the steady ABC flow defined in equation 3.15. The particle size parameter was $S = 10^{-5}$ and the density ratio was $R = 10^{-3}$.

3.8 Considering the history force

In the second chapter it was shown, as has been found in the literature [7, 31, 72, 73], that the history force plays a significant role in particle behaviour for some regions of the parameter space. In this section, we demonstrate that the shape of the inertial manifold changes when history is included for the small density ratio case of sand in water.

We simulated 10^5 inertial particles in a fluid velocity field given by equation 3.11, with and without history. As before, the particle positions have been binned over a regular cartesian grid. After binning, the average velocity difference between the particles and the inertial manifold has been calculated for each cell. The velocity difference between Lagrangian particles without

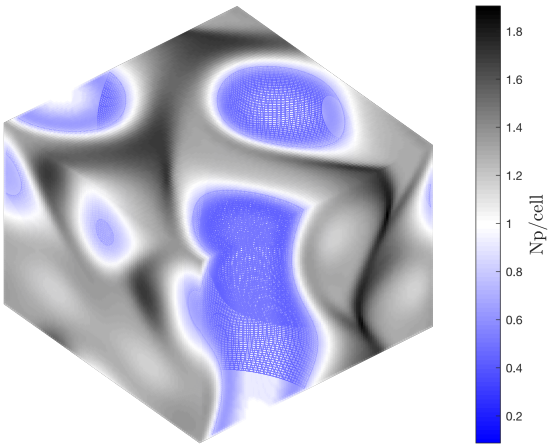


Figure 3.24: *An advantage of a transfer operator approach is that long term system dynamics can be found by calculating the eigenvectors of W_{ij}. The long term particle spatial density is shown in a box of $2\pi\mathrm{m}^3$ with a fluid velocity field described by the ABC flow, with $A = 1$, $B = \sqrt{2}$ and $C = \pi$. The particle parameters are $S = 10^{-5}$ and $R = 10^{-3}$. The colour scale represents the number of particles in each cell; initially, each grid cell contained 1 particle. Particles are ejected from vortices, which can be seen as blue tubular structures.*

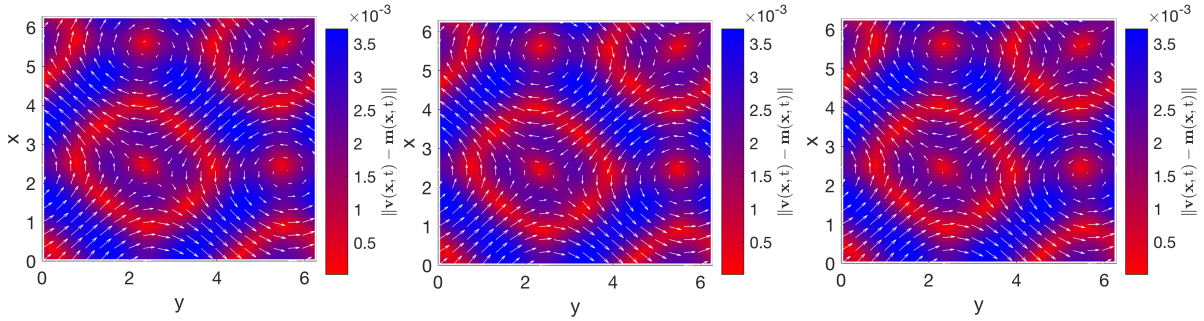
history is given by $\|\widehat{\mathbf{v}} - \mathbf{m}\|(\mathbf{x}, t)$, and the average binned velocity difference including history is given by $\|\widehat{\mathbf{v}}_h - \mathbf{m}\|(\mathbf{x}, t)$. Because the computational requirements of including history are large, a smaller number of particles have been simulated. Some cells contained no particles when the averages were taken and resulted in a small amount of noise in the surfaces.

We consider here a special case of particle advection including the history force in which the particles are released at the local velocity of the fluid. The influence of history, therefore, starts from zero and grows over time as the historical tails of the particles grow. It is therefore expected that the shape of the particle velocity surface, $\|\widehat{\mathbf{v}}_h - \widehat{\mathbf{m}}\|(\mathbf{x}, t)$ will change over time.

Figures 3.25 and 3.26 demonstrate that when history is required, the inertial equation does not necessarily provide accurate initial conditions for the particle velocities. In order to extend the use of a transfer operator to cases where history is important, a method must be devised that discretises the historical behaviours of the particles in the same way that we have discretised the particle phase space.

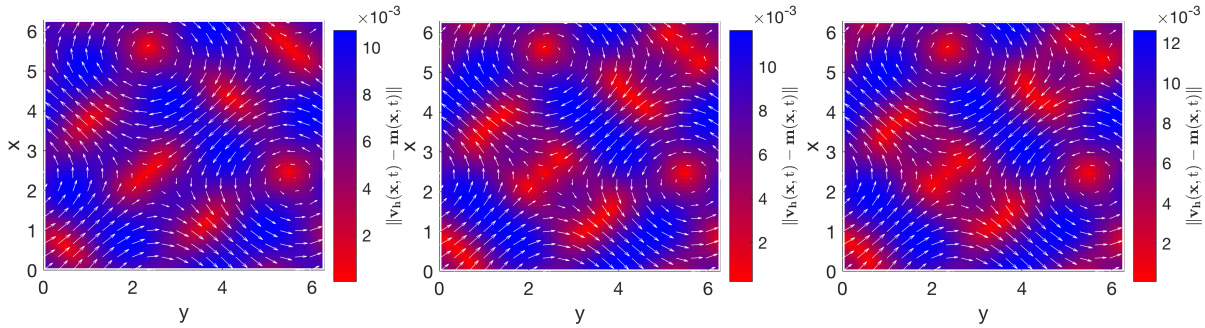
3.9 Conclusions

We have successfully demonstrated that a transfer operator approach can be used to describe the dynamics of inertial particles. A linearisation of a transfer operator, W_{ij} was first constructed that discretised the entire particle phase space, but discretising the entire particle phase space was



(a) *after* $0.12T$ ***(b)*** *after* $0.36T$ ***(c)*** *after* $0.72T$

Figure 3.25: *The difference in velocity between Lagrangian particles described by equation 3.1 and the velocity described by equation 2.13, reaches an approximately constant value after* $0.12T$*. Each subfigure shows the same shape and magnitude of the velocity difference. The particle parameters were given by* $S/R = 0.0419$ *with* $R = 0.4$*.*



(a) *after* $0.12T$ ***(b)*** *after* $0.36T$ ***(c)*** *after* $0.72T$

Figure 3.26: *The velocity difference between the Lagrangian history tracers and the inertial manifold evolves over time. It implies that either the shape of the manifold is unsteady in the case of the history tracer, or that the attraction towards an inertial manifold is much slower when history is included. Again* $S/R = 0.0419$ *with* $R = 0.4$*.*

algorithmically inefficient. By reducing the dynamics of the particles onto an inertial manifold we can produce a more computationally tractable approach.

However, reducing the dynamics onto an inertial manifold causes the accuracy of our method to become dependent on the parameter S/R. The parameter S/R governs the degree to which particles follow the fluid. We have been able to identify the region of the parameter space for which our efficient approach is valid, by investigating the dynamics of Lagrangian particle trajectories with respect to the small parameter, S/R.

For $S/R < 0.1$, clustering is slightly under predicted by W_{ij} because the first order remapping approach suffers from numerical diffusion. For larger S/R, we showed that particle behaviour fundamentally changes, and the separatrix running between vortices in our test flow is no longer the only attracting orbit. Particles are attracted to a band of orbits that run either side of the separatrix, a behaviour that is not captured by our discretisation of the transfer operator.

The remapping approach we used is approximately first order with the grid spacing parameter Δx, as was shown in figure 3.22. We deliberately used a simple remapping procedure to allow W_{ij} to be constructed in higher dimensions.

Our method has advantages over simulating large numbers of Lagrangian particles. Far fewer particles are required to reach statistical convergence when a transfer operator approach is utilised. In steady flows, our approach is incredibly efficient, because the long term dynamics of the system are captured by the eigenvectors of W_{ij}. Further an Eulerian description of particles allows us to model particle collisions very efficiently, as we shall see in the next chapter. Feed back on the fluid can also be modelled effectively using an Eulerian approach, something we would like to explore in future work.

In the final section we showed that the history force changes the shape of the surface onto which particle trajectories are attracted. For certain values of S and R, specifically larger values of both, the requirement to include history for accurate modelling means that the transfer operator approach cannot be used. However, we identified clearly the regions for which history was important in the second chapter. By adhering to those findings we know exactly by how much not including history is affecting our results.

Finally we remark that our method is certainly effective in some cases, but that care should be taken with regard to the values of S and R before it is employed in earnest.

CHAPTER 4

COLLISION DETECTION

4.1 Introduction

In some instances, interactions between particles are an important concern. For example, in raindrop formation in warm clouds, collisions between droplets of water are an important mechanism by which raindrops form [15]. Any process which involves large numbers of colloidal particles is expensive to simulate computationally. The complexity arises from the requirement to conduct distance comparisons between each possible particle pair. For a simulation containing N particles, roughly N^2 operations must be performed each time step to test for collisions.

In this chapter, we investigate methods of collision detection that have better algorithmic complexity than order N^2. These range from methodologies that exploit particle concentration fields to implicitly map collisions, to those that subtly modify explicit collision detection algorithms to reduce complexity. Reducing algorithmic complexity without compromising accuracy allows a wider range of natural processes to be modelled and understood.

We compare three methods: the cell method, which improves on pairwise collision detection through subdivision of the domain into smaller cells; the strain tensor approach, an apparently novel approach which makes use of eigenvalues of the Cauchy-Green strain tensor to track volumes around trajectories; and the transfer operator method, which approximates collisions probabilistically based on the local concentration of particles. Each method is compared in terms of predicted collision locations and times, and algorithmic complexity. We also provide general comments on the difficulty of implementation. Our key finding is that both the transfer operator method and strain tensor approaches are able to model collisions with algorithmic complexity independent of the number of particles in the simulation.

For each method we conduct particle simulations in the 2-dimensional test flow utilised in

the previous chapters. The flow consists of two periodic vortices and is given by,

$$\mathbf{u}(\mathbf{x},t) = \begin{bmatrix} A\cos y + B\sin y \\ B\cos x + A\sin x \end{bmatrix}, \tag{4.1}$$

where A and B are constants that govern the vortex symmetry and characteristic velocity of the flow.

4.2 Collision detection methods

4.2.1 Collisions through direct numerical simulation: The cell method.

A straightforward method for modelling collisions is to compare particle positions every time step. A collision is an instant in which two particles are closer together than the sum of their radii. Consider two particles, particle A and particle B, with positions p_A and p_B, and diameters d_A and d_B. A collision is defined as an instance that satisfies $\|\mathbf{p}_A - \mathbf{p}_B\| < (d_A + d_B)/2$, where $\|\mathbf{p}_A - \mathbf{p}_B\|$ is the Euclidian distance between the two particles. Such an approach requires the calculation of pairwise distances between particles each time step. When conducted naively, this method of collision detection has algorithmic complexity of $O(N^2)$, where N is the number of particles in the simulation. The N^2 complexity arises from requirement to compare position between each possible particle pair. The algorithmic complexity can be significantly improved using the cell method first suggested in [87].

We will refer to the space of particle positions as the “configuration space", which may be 1, 2, or 3 dimensional. In the cell method, the configuration space of the particles is divided into m equal and overlapping cubes as in figure 4.1. The small cubes must overlap by at least 1 particle diameter to ensure that collisions between particles in adjacent grid cells are included. Collisions are then detected only between particles within each of the sub-cells, reducing the algorithmic complexity.

Algorithmically, the process is as follows. First, each particle is assigned to a cell based on its position in space, which requires $N \times m$ operations. Then, collisions are checked for within each cell. If we assume that particles are homogeneously distributed in space then the number of particles in each cell is equal to N/m. Therefore, the number of operations required to check for collisions within a single cell is equal to $(N/m)^2$. For a domain spilt into m cells, we require $m \times (N/m)^2$ operations to check for collisions across the whole space.

In total, the algorithm requires $N \times m$ operations for assigning particles to cells and an additional $m \times (N/m)^2$ operations to check for collisions within each cell. The total number of operations required by the algorithm is given by,

$$N_{operations} = N \times m + m \times (N/m)^2. \tag{4.2}$$

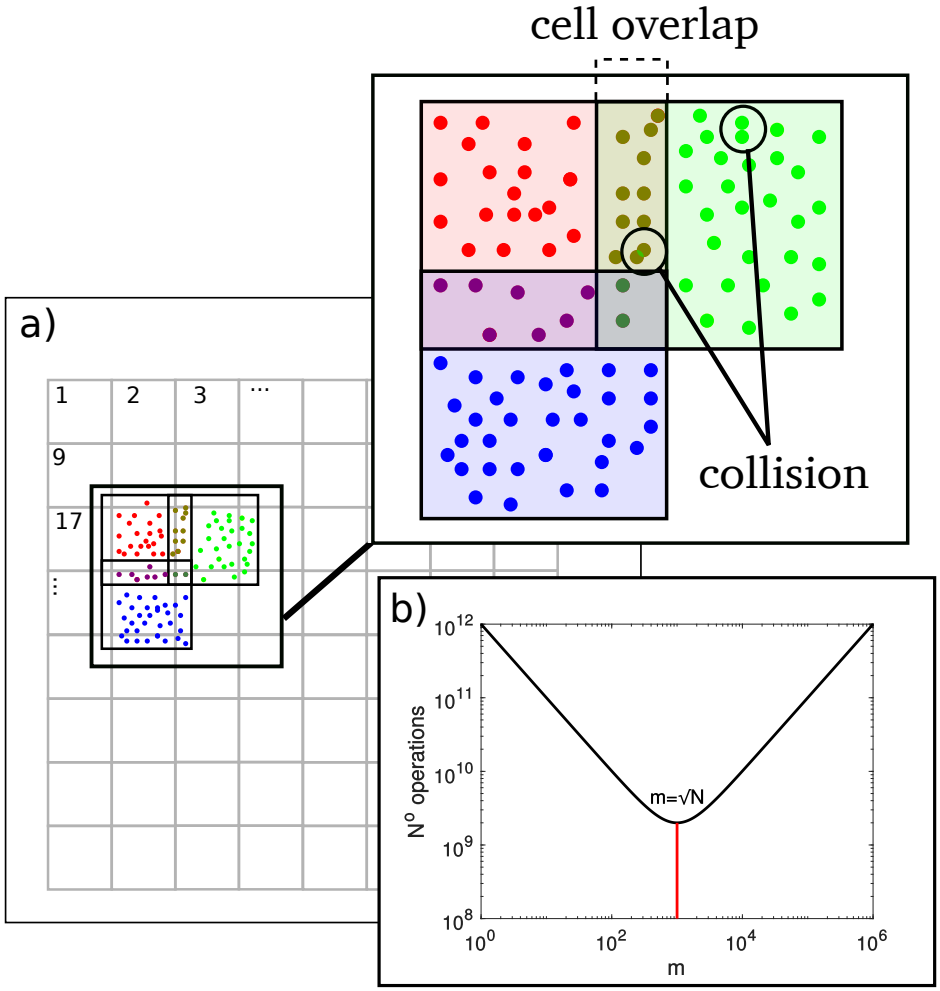


Figure 4.1: *The first algorithm we investigate for collision detection is the cell method. Configuration space is divided into overlapping cells as is shown in (a). Collisions are then searched for between particles in each of the cells. The cell method has considerably more favourable algorithmic complexity than pair wise collision detection. Through careful choice of the number of cells m, we can minimise the number of operations required to detect particle collisions. A numerically optimum choice for m is given by $m = \sqrt{N}$, which can be seen in b).*

We wish to minimise the required number of operations through tuning of the parameter m. Through simple differentiation,

$$\begin{aligned} \frac{dN_{operations}}{dm} &= N - \frac{N^2}{m^2} \\ 0 &= N - \frac{N^2}{m^2} \\ m &= \sqrt{N}, \end{aligned} \tag{4.3}$$

we can see that the most efficient choice for m is $\sqrt{N}$, a well known result [88]. The algorithmic efficiency is plotted as a function of m in figure 4.1b). Clearly, sensible choice of m leads to

dramatic improvements in efficiency. For an ideal distribution of particles in space, i.e. particles distributed evenly amongst the m sub cells, we can calculate the algorithmic complexity of the cell method. Substituting the optimal number of boxes, $m = \sqrt{N}$, into equation 4.2, leads to $N_{operations} = N\sqrt{N} + \sqrt{N} \times (N/\sqrt{N})^2 = \sqrt{N} + \sqrt{N} \times N$ which scales with $N^{3/2}$.

A further concern is the time frequency with which collisions are checked. When collisions are detected by comparing instantaneous positions, particles cannot be allowed to move by more than one particle diameter between each collision sweep: two particles may pass through one another within one time step and the collision be missed. Recall the two particles, Particle A and particle B. Their positions are given by $\mathbf{p}_A$ and $\mathbf{p}_B$, their diameters by d_A and d_B, and their velocities by $\mathbf{v}_A$ and $\mathbf{v}_B$. For simplicity let us assume that between time steps particles travel along straight paths. In order to capture a collision between particle A and particle B, the component of their relative velocities along the line of their centres, $\frac{(\mathbf{v}_A - \mathbf{v}_B)\cdot(\mathbf{p}_A - \mathbf{p}_B)}{\|\mathbf{p}_A - \mathbf{p}_B\|}$, must satisfy,

$$\frac{(\mathbf{v}_A - \mathbf{v}_B)\cdot(\mathbf{p}_A - \mathbf{p}_B)}{\|\mathbf{p}_A - \mathbf{p}_B\|}\Delta t < d_a + d_b, \tag{4.4}$$

where Δt is the time between sweeps.

In our particular application of the algorithm, there are conditions on the velocities of the particles. The conditions on the particle velocities mean that for certain parameter regimes missed collisions will not be a problem. Conditions on the particle velocity difference are imposed by the particle Reynolds number, Re_p, which must remain small. In the first chapter, with motivation from the literature, we chose this particle Reynolds number limit to be,

$$Re_p = \frac{\rho_f \|\mathbf{v} - \mathbf{u}\| d_a}{\mu} < 5. \tag{4.5}$$

In the first chapter, we also investigated the parameter space defined by the particle size parameter and the density ratio ρ_p/ρ_f. In the parameter space we indicated the set of parameters that caused the particles to exceed the limit $Re_p > 5$. We found that in the case of aerosol particles, $\rho_p > \rho_f$, the line of limiting particle Reynolds number, i.e. the line through the parameter space that defines the region $Re_p > 5$, is approximately given by $\rho_p/\rho_f = 0.0045S^{-1.4038}$. Choosing parameters that represent water droplets in air from the line of maximum Re_p, we can calculate the maximum possible velocity difference, $\|v_a - v_b\|$. Choosing $\rho_f/\rho_p = 1000$, the corresponding size parameter is 1.5445×10^{-4}. The size parameter is given by, $S = \frac{d_a^2 \rho_f}{18\mu}$, where d_a is the diameter of the particle. We non-dimensionalise it with characteristic length L and velocity U_0 to give $\widehat{S} = \frac{d_a^2 \rho_f U_0}{18L\mu}$. The particle diameter is therefore given by,

$$d_a = \sqrt{\frac{18L\mu\widehat{S}}{U_0\rho_f}}, \tag{4.6}$$

substituting 4.6 into 4.5 gives,

$$Re_p = \|\mathbf{v} - \mathbf{u}\| \sqrt{\frac{\rho_f 18L\widehat{S}}{U_0\mu}} < 5 \tag{4.7}$$

For the test flow in equation 4.1, with $A = 1$ and $B = \sqrt{2}$, and the example of water droplets in air, $L = 2\pi$, $U_0 = 1.66$ and $\mu = 1.57 \times 10^{-5}$. Substituting these and our previously calculated value of $\widehat{S}$ into 4.7 we calculate that,

$$\|\mathbf{v} - \mathbf{u}\| < 0.1931, \tag{4.8}$$

is the maximum velocity difference that satisfies $Re_p < 5$. Taking the worst case that $v_a - u = 0.1931$ and $v_b - u = -0.1931$, the maximum difference between v_a and v_b that satisfies all our constraints is $v_a - v_b < 0.3863$. With these parameters, $d_a = 4.0646 \times 10^{-4}$. Recalling our constraint for identifying all collisions, $\frac{(\mathbf{v}_A - \mathbf{v}_B)\cdot(\mathbf{p}_A - \mathbf{p}_B)}{\|\mathbf{p}_A - \mathbf{p}_B\|}$ we can now calculate Δt required to capture all collisions. Assuming again the worst case that the velocity difference acts along the line of centres of the two particles, $\Delta t < 0.0021$. Although this approach is not mathematically rigorous, it effectively demonstrates that, even with our constraints on the particle velocities, it is possible for collisions to be missed between time steps. It is therefore prudent to take steps to eliminate missed collisions.

To improve our approach we assume that between time steps particles follow linear paths. The trajectories of our two particles within the next time step are given by,

$$\begin{aligned} \mathbf{p}_A(t_0 + \Delta t) &= \mathbf{p}_A(t_0) + \Delta t \mathbf{v}_A(t_0) \\ \mathbf{p}_B(t_0 + \Delta t) &= \mathbf{p}_B(t_0) + \Delta t \mathbf{v}_B(t_0), \end{aligned} \tag{4.9}$$

where t_0 is the time at the beginning of the time step. To search for collisions between time steps, we look for instances in which,

$$\begin{aligned} &0 < t < \Delta t \\ &\|\Delta x + t\Delta v\| < (d_A + d_B)/2, \end{aligned} \tag{4.10}$$

where t is the time relative to the start of the timestep, $\Delta\mathbf{x} = \mathbf{p}_A(t_0) - \mathbf{p}_B(t_0)$ and $\Delta\mathbf{v} = \mathbf{v}_A(t_0) - \mathbf{v}_B(t_0)$. Squaring both sides gives,

$$\|\Delta\mathbf{x}\|^2 + \|\Delta\mathbf{v}\|^2 t^2 + 2\Delta\mathbf{v}\cdot\Delta\mathbf{x}t - \frac{(d_A + d_B)^2}{4} < 0, \tag{4.11}$$

which can be solved for t. If there are solutions of 4.11 which satisfy $0 \le t \le \Delta t$, then a collision will take place in the next time step.

Our approach assumes that between time steps the inertial particles follow linear paths. Whilst the collision detection scheme is first order, compared to the second order integration scheme, we are confident that results will be unaffected by the simplification. Differences in collision rates between particles tracing linear and quadratic paths are thought to be minor for the small time step used. Sigurgeirsson et al. [88] provide further methods for improving between time step collision detection by also considering the particle acceleration between time steps. For our purposes, the assumption of linear paths between time steps is sufficient.

4.2.2 Collisions using the eigenvalues of the Cauchy-Green strain tensor.

Particle collision rates can be approximated based on local particle concentration. Here we demonstrate an approach for modelling collisions based on the rate at which nearby particle trajectories diverge from one another. Regions of a flow in which particles move away from one another are unlikely to produce collisions. Conversely, particles are likely to collide in regions in which particle trajectories move towards one another. A commonly used measure of the nature of attractors in dynamical systems are Lyapunov exponents. Lyapunov exponents measure the average growth rate of small perturbations to orbits around an attractor. Large positive Lyapunov exponents indicate a system is chaotic and negative exponents indicate predictable attraction to an equilibrium point or an orbit.

In our case, we are interested in the relative behaviour of nearby particle trajectories at finite time instances. We use an approach similar to finite time Lyapunov exponents (FTLE), which denote the exponential rate at which two nearby trajectories diverge. Initially, two points may be infinitesimally close, but as the flow evolves, the distance between the points can shrink or grow. finite time Lyapunov exponents (FTLE) are commonly used for identifying transport barriers in flows [89]. Ridges in the FTLE field indicate barriers between regions which are seldom crossed. Two points that start either side of a transport barrier will move away from one another, resulting in large positive FTLE in that location.

Given assumptions about initial distances between trajectories, the FTLE can also be used to query the distance between trajectories at a specific instance in time and space. Knowledge of the distance between trajectories provides a tool for detecting collisions.

In the proceeding section we introduce the Cauchy-Green strain tensor, the eigenvalues of which provide information about the stretch and compression around a trajectory much like the FTLE. We show how the Cauchy-Green strain tensor can be used for collision detection and extend the method to apply to the MR equation with history. Our approach provides an efficient method for investigating the influence of the Basset history force on the divergence rate of nearby particle trajectories, and a method for modelling particle collisions.

We consider the flow map F_{t_0}, which may represent the active tracer, history tracer or passive tracer systems. Two particle trajectories can be described by, $\hat{x}(t) = F^t_{t_0}(x_0)$ and $\hat{y}(t) = F^t_{t_0}(y_0)$. If initially the distance between the two particles is $\delta x_0 = \|\hat{x}(0) - \hat{y}(0)\|$, then the particles can be approximately expected to diverge such that,

$$\|\delta x_t\| \approx \|\delta x_0\| e^{\Lambda(t-t_0)}, \tag{4.12}$$

where the exponent Λ is the the exponential divergence rate we wish to calculate.

Initially the positions of the particles are related by $y_0 = x_0 + \delta x_0$. Therefore, under the flow map $F^t_{t_0}$, the distance between $\hat{x}(t)$ and $\hat{y}(t)$ at time t is given by $\delta x(t) = F^t_{t_0}(x_0 + \delta x_0) - F^t_{t_0}(x_0)$.

Expanding δx_t about $\delta x_0 = 0$ gives,

$$\begin{aligned}\delta x(t) &\approx F^t_{t_0}(x+\delta x_0) - F^t_{t_0}(x) + \nabla F^t_{t_0}(x+\delta x_0)\delta x_0 + O\delta x^2 \\ &\approx \nabla F^t_{t_0}(x)\delta x_0,\end{aligned} \tag{4.13}$$

and taking the Euclidian norm gives,

$$\begin{aligned}\|\delta x(t)\| &\approx \|\nabla F^t_{t_0}(x)\delta x_0\| \\ &= \sqrt{(\delta x_0 \nabla F^t_{t_0}(x)) \cdot (\nabla F^t_{t_0}(x))\delta x_0} \\ &= \sqrt{\delta x_0 \cdot (\nabla F^t_{t_0}(x))^T (\nabla F^t_{t_0}(x))\delta x_0}\end{aligned} \tag{4.14}$$

For simplicity in notation we now introduce the symmetric matrix, $\mathbf{C}(x_0, t_0, t) = (\nabla F^t_{t_0}(x))^T(\nabla F^t_{t_0}(x))$, known as the right Cauchy-Green strain tensor, where superscript ^{T} denotes matrix transposition. The Cauchy-Green strain tensor gives the growth factor of the initial distance δx_0 after $t - t_0$ seconds. To find the magnitude of maximum stretch in one direction, we find $\max\left(\sqrt{\delta x(0) \cdot \mathbf{C}\delta x(0)}\right) = \|\delta x(0)\|\sqrt{\lambda_{max}(\mathbf{C})}$. Rearranging this gives,

$$max\frac{\|\delta x(t)\|}{\|\delta x(0)\|} = \sqrt{\lambda_{max}(\mathbf{C})}. \tag{4.15}$$

By recalling our initial statement that the exponential growth rate, Λ, must satisfy $\|\delta x_t\| \approx \|\delta x_0\| e^{\Lambda(t-t_0)}$, then we can see how Λ must relate to C. If,

$$\|\delta x_t\| \approx \|\delta x_0\| e^{\Lambda(t-t_0)} = \|\delta x_0\|\sqrt{\lambda_{max}(\mathbf{C})} \tag{4.16}$$

then we can see that,

$$\Lambda = \ln(\sqrt{\lambda_{max}(\mathbf{C})})/(t - t_0). \tag{4.17}$$

So to summarise, information about the expansion of the volume surrounding a trajectory is contained within $\mathbf{C}(x_0, t_0, t)$. The magnitude and direction of maximum stretch is given by the maximum eigenvalue and corresponding eigenvector of $\mathbf{C}$. To calculate the exponential growth rate we take logarithms of the eigenvalues of $\mathbf{C}$ and normalise by the integration period $t - t_0$.

In practice, we calculate $\mathbf{C}$ by first calculating the tangent space of the original system. The tangent space sits tangent to the velocity surface given by the original system, and has dimensionality equal to the number of velocity dimensions. By investigating how vectors in the tangent space deform, we can construct a matrix equivalent to $\mathbf{C}$. The principle can be readily visualised for a 1−dimensional flow.

As an example we choose $\dot{x} = \cos(x)$. The tangent space is found by linearising the original velocity field. In this case, the tangent space equation is given by $\dot{\delta x} = -\sin(x)\delta x$, where δx represents a small perturbation away from a trajectory in the original system. We now have a coupled system consisting of the tangent space and the original system, given by,

$$\begin{aligned}\dot{x} &= \cos(x) \\ \dot{\delta x} &= -\sin(x)\end{aligned} \tag{4.18}$$

Solving the coupled system from the point $(\delta x_0, x_0)$ not only gives a trajectory in the original system, but also indicates how trajectories close to it behave.

Figure 4.2 demonstrates how a trajectory and the corresponding tangent space behave in the flow $\dot{x} = \cos(x(t))$. A trajectory is evolved from the point $x(0) = 0$, and alongside it, $\delta x(t)$ is evolved from the point $\delta x(0) = 1$. As the trajectory approaches the attractor at $x(t) = \pi$, $\delta x(t)$ approaches zero. Similarly, the value of the exponential divergence rate Λ approaches -1. Essentially, Λ is telling us that trajectories move closer together at the attractor, which is exactly what one would expect.

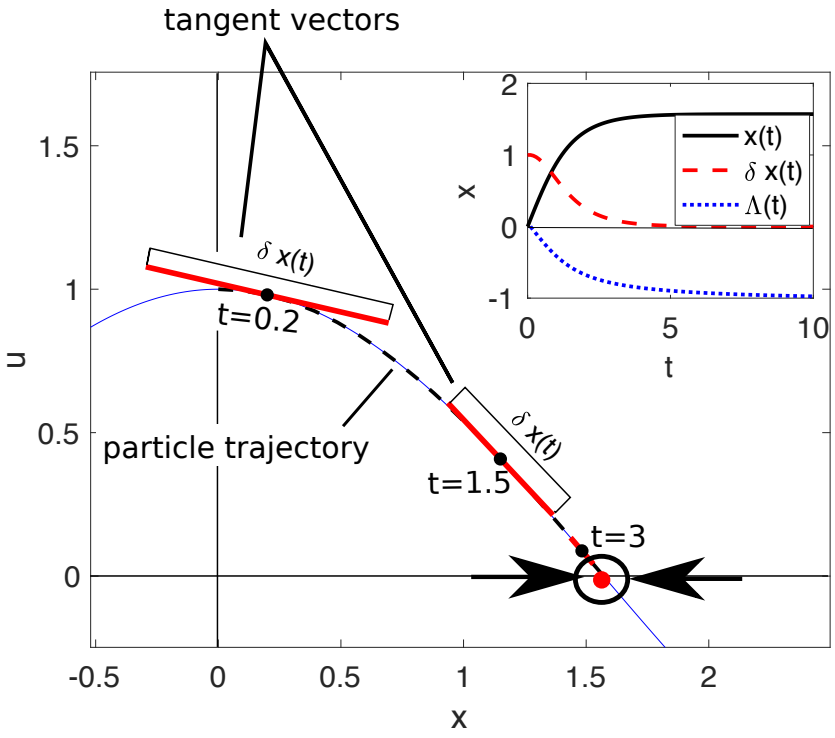


Figure 4.2: *The eigenvalues of the Cauchy-Green strain tensor indicate the contraction of volumes surrounding particle trajectories. An attractor is present in the velocity field given by $u = \cos(x)$. As a trajectory (dashed black line) approaches the attractor, δx shrinks, indicating that trajectories move closer together. inset: we can see that as the trajectory reaches the attractor at $x = \pi/2$, δx approaches zero, whilst $\Lambda(t) = \frac{1}{t}\ln(\delta x(t))$ approaches -1. Negative values of Λ indicate attractive regions.*

In higher dimensions, the tangent space is used to evolve a set of orthonormal basis vectors and produce a matrix equivalent to $\mathbf{C}$. The eigenvalues of $\mathbf{C}$ are used to calculate what is known as the Lyapunov spectrum. To demonstrate the process we use the example of a passive tracer

particle in the steady ABC flow, given by

$$\mathbf{u}(x) = \begin{bmatrix} \widehat{C}\cos(y) + \widehat{A}\sin(z) \\ \widehat{A}\cos(z) + \widehat{B}\sin(x) \\ \widehat{B}\cos(x) + \widehat{C}\sin(y) \end{bmatrix}, \tag{4.19}$$

where, $\widehat{A}$, $\widehat{B}$ and $\widehat{C}$ govern vortex symmetry and fluid velocity.

As before, the tangent space describes a linearisation of the velocity field that can be integrated alongside the particle equations of motion. The linearisation around the ABC flow is given by,

$$\dot{Z} = \nabla\mathbf{u}Z \quad \text{where} \quad \mathbf{Z} = (\delta x, \delta y, \delta z) \tag{4.20}$$

and,

$$\nabla\mathbf{u} = \begin{bmatrix} 0 & -C\sin(y) & A\cos(z) \\ B\cos(x) & 0 & -A\sin(z) \\ -B\sin(x) & C\cos(y) & 0 \end{bmatrix} \tag{4.21}$$

where $\dot{Z}$, are the tangent space equations: a set of first order linear ODEs that describe a linearised velocity field $Z = \lim_{\delta\mathbf{x}\to 0}(\mathbf{x} + \delta\mathbf{x})$, close to the original passive tracer system [90]. We are interested in how small perturbations away from a trajectory behave with respect to each basis direction of the coordinate system. We therefore use the tangent space to evolve each of the initial perturbations, $e_x = (1,0,0)$, $e_y = (0,1,0)$ and $e_z = (0,0,1)$ alongside a passive tracer trajectory. In total a system of 12 coupled equations are integrated numerically, the solutions of which construct a matrix of the form,

$$\mathbf{M}(\mathbf{x},t) = \begin{bmatrix} [\delta x(t)]_x & [\delta x(t)]_y & [\delta x(t)]_z \\ [\delta y(t)]_x & [\delta y(t)]_y & [\delta y(t)]_z \\ [\delta z(t)]_x & [\delta z(t)]_y & [\delta z(t)]_z \end{bmatrix},$$

where the subscripts x, y and z indicate the directions of the perturbation with which the tangent space was initialised. For example, the top row of $\mathbf{M}$ shows how the tangent space evolves with a perturbation in the e_x direction. $\mathbf{M}$ is therefore equivalent to $\nabla F^t_{t_0}(x)$ and the eigenvalues of the matrix $\mathbf{M}^T\mathbf{M}$ can be used to approximate the exponential divergence rate Λ.

Figure 4.3 shows a planar slice through $\Lambda(\mathbf{x})$, calculated for a field of points using equations 4.21. Evolution of the tangent space was conducted for 0.7240 characteristic timescales and the planar slice was given by $z = \pi$. The ABC flow contains toroidal vortices between which run a series of chaotic trajectories. Ridges in $\Lambda(\mathbf{x})$ pick out the boundaries of the toroidal vortices that define boundaries through which there is no fluid transport. However, inertial particles within the ABC flow will be ejected from the vortices and concentrate in the chaotic region between them.

Including inertia complicates the calculation of the matrix $\mathbf{M}$. Because the active tracer is a second order system, the entire phase space should be considered in the calculation of $\mathbf{M}$. The

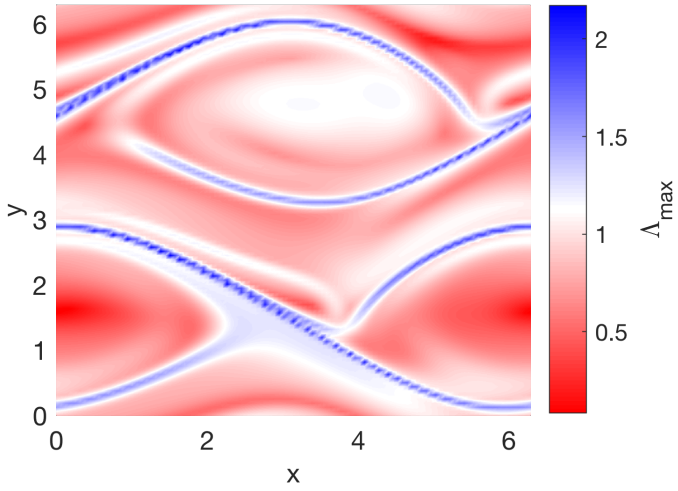


Figure 4.3: *Plotting a planar slice through a field of* Λ *values reveals transport barriers in the ABC flow. Passive tracer trajectories were calculated backwards in time for* 0.7240 *characteristic timescales from* $z = \pi$ *for a well resolved grid of* (x, y) *positions. The blue ridges mark the boundaries between vortices in the ABC flow.*

matrix M becomes a $d \times d$ matrix where d is the number of dimensions in the system's phase space. When history is included, the phase space of the system becomes infinite dimensional. The current state in the system is dependent on a continuum of previous states. Therefore, the tangent space of the full MR equation is also infinite dimensional and is given by,

$$\begin{aligned}
\delta\dot{\mathbf{x}}_1 &= \delta\mathbf{x}_2 \\
\delta\dot{\mathbf{x}}_2 &= \frac{3}{2}R\nabla\left(\frac{D\mathbf{u}}{Dt}\right)\delta\mathbf{x}_1 - \frac{R}{S}(\delta\mathbf{x}_2 - \nabla(\mathbf{u})\delta\mathbf{x}_1) \\
&\quad - 3R\sqrt{\frac{1}{2\pi S}}\int_{-\infty}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\delta\mathbf{x}_2 - \nabla(\mathbf{u}(\tau))\delta\mathbf{x}_1)d\tau,
\end{aligned} \tag{4.22}$$

where,

$$\mathbf{x}_1 = \begin{vmatrix} x \\ y \\ z \end{vmatrix}$$

Equation 4.22 can be reduced to a finite dimensional equation and solved using the numerical method in the second chapter. The basset integral is discretised into a vector of previous system states at discrete intervals in time. To construct the matrix M in full, we require a set of tangent space equations for each entry in the vector of previous system states. Such a matrix is extremely large. Fortunately we are only interested in the relative behaviour of particles in configuration space. Because each set of tangent space equations are independent of one another, we are required only to consider certain columns of the matrix M. Columns which do not correspond to

initial perturbations in configuration space can be ignored and the exponential divergence rate Λ for the full MR equation can be calculated.

We now demonstrate how the Cauchy-Green strain tensor can be used for detecting collisions between particles. We will refer to this approach as the strain tensor approach. As we have discussed, the exponential divergence rate Λ can be used to track the compression of the velocity field surrounding trajectories. By making assumptions about the initial particle distributions, it is possible to use the degree of compression to indicate likely collision locations and rates. To model collisions using Λ, we assume that initially particles are homogeneously distributed in configuration space. Collisions will occur when there has been enough compression around a trajectory for a particle to collide with its nearest neighbour.

We assume that the distances to a particle's nearest neighbour are distributed like a spatial poisson process. The distance to a particle's nearest neighbour is then given by,

$$P(r) = \begin{bmatrix} 2\rho\pi r \exp(-\rho\pi r^2) & n = 2 \\ 4\rho\pi r^2 \exp(-\frac{4\rho\pi r^3}{3}) & n = 3 \end{bmatrix} \tag{4.23}$$

where r is the radial distance from a particle's centre, n is the number of dimensions, and ρ is the spatial density of particles [91]. Initially, we assume that the distance of any particle to its nearest neighbour, is given by the mean of $P(r)$. The mean of $P(r)$, which we denote with σ, is the expected distance between a randomly selected particle and its nearest neighbour.

Given assumptions about the initial distance between particles, we can define a collision condition. For a collision to occur a particle of radius r_p must be $\frac{2r_p}{\sigma}$ times closer to its neighbours than it was at the start of the simulation to result in a collision. The question is, how many times closer together are particles at time t?

As we know, the compression around trajectories in phase space is given by the eigenspectrum of $\mathbf{M}(t_0, x_0, t)$. In this case however, we are interested in how trajectories approach each other only in configuration space. As such, we do not calculate Λ for the full 6-dimensional phase space. Instead we calculate the compression in configuration space alone. A matrix is constructed from small perturbations evolved by the tangent space equations, as before, but the perturbations are only defined in the spatial dimensions. We denote the matrix containing the subsample of tangent space solutions as $\widehat{\mathbf{M}}(x_0, t_0, t)$. The square root of the determinant of $[\widehat{\mathbf{M}}^T\widehat{\mathbf{M}}]$ at time t, tells us the scale factor on the volume surrounding a trajectory.

To help us illustrate the idea let us consider an example. In two dimensions, a tangent space can be imagined as a plane tangent to a surface, as in figure 4.4. The size of the initial perturbations denote the initial volume of the tangent space. After some time $t_1 - t_0$, the initial perturbations have been deformed as they followed a trajectory over the surface. The product of the eigenvalues, λ_1 and λ_2, are the square root of the determinant of $[\widehat{\mathbf{M}}^T\widehat{\mathbf{M}}]$ and give the scale factor on the volume surrounding the trajectory. If we assume for simplicity, that the distance between particles changes by the n^{th} root of the volume scale factor, where n is the number of

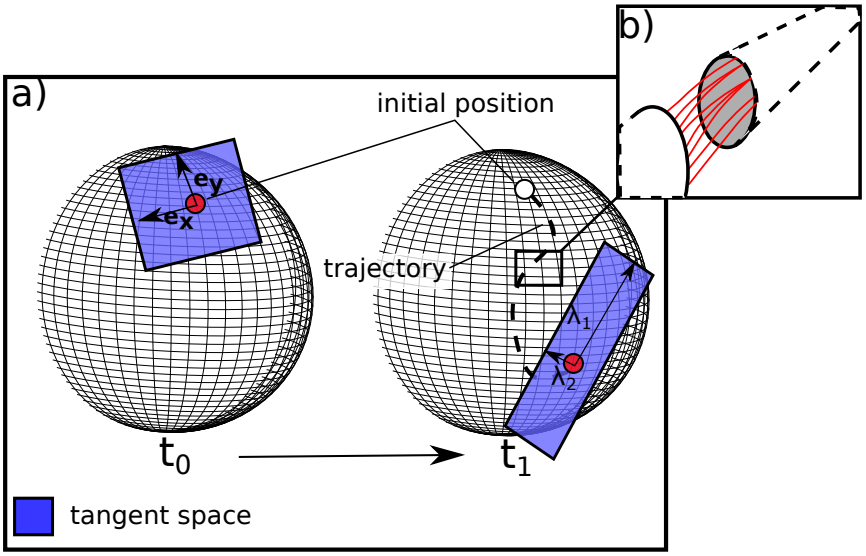


Figure 4.4: *In higher dimensions the tangent space can be used to track volumes surrounding trajectories. a). A particle traces a trajectory on the surface of a sphere. As the particle moves, the size and shape of the tangent space changes. At time t_1 the original perturbations have deformed. The product of the eigenvalues λ_1 and λ_2, give the scale factor between the new and old shapes. b) Although the compression has been mapped around only one trajectory, within it can be imagined a host of trajectories that converge or diverge. Using one trajectory to represent many increases computational savings.*

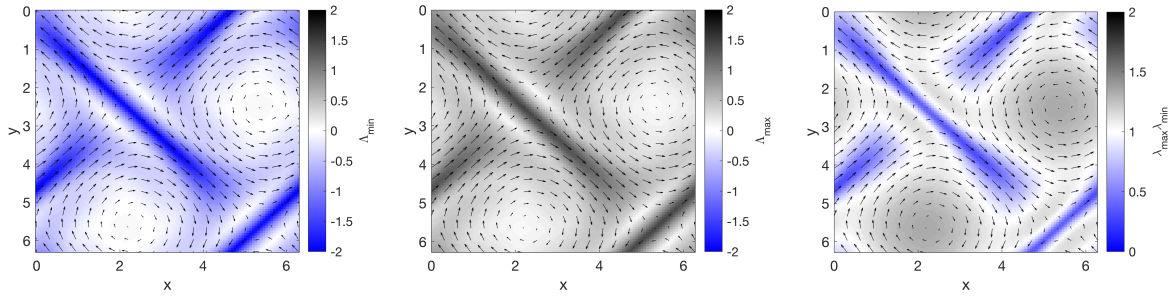
dimensions, then a collision occurs when the following condition is satisfied,

$$\left(\sqrt{\det\left[\widehat{\mathbf{M}}^T\widehat{\mathbf{M}}\right]}\right)^{\frac{1}{n}} \leq \frac{2r_p}{\sigma}. \tag{4.24}$$

We found that it is important to consider the change in volume predicted by the tangent space, rather than considering only maximal compression associated with the largest eigenvalue of $\mathbf{M}^T\mathbf{M}$. When only maximal compression in one direction was considered, our approach overestimated the collision rates between particles. The collisions were assumed to occur when the eigenvalues of the Cauchy-Green strain tensor indicated sufficient compression in one principle direction, regardless of the stretching in another. Whilst particles were closer together in that direction, they were often further away in another. This point is illustrated by figure 4.5.

Figure 4.5 (a) and (b) show the maximum and minimum values of Λ, Λ_{min} and Λ_{max}, calculated from the Cauchy-Green strain tensor. The simulation was run backwards in time using the active tracer and the flow field given in equation 4.30 introduced in the following section. Figure 4.5(c) shows the product of the two eigenvalues of $\mathbf{M}^T\mathbf{M} = \lambda_{max} \times \lambda_{min}$ for the same simulation. We have chosen colours which make the compression and expansion regions clear. Blue indicates that a region is compressive and dark grey indicates the region is expansive. Recall that $\Lambda_{max} = \ln(\lambda_{min})/(t_1 - t_0)$, where $(t_1 - t_0)$ is the integration period. Therefore regions of compression correspond to negative values of Λ. The separatrix along which Λ predicts collisions,

is the region of greatest compression in the direction perpendicular to the separatrix. However, it is also the region of greatest stretch. The stretch runs parallel to the separatrix. The particle field is compressed in the direction perpendicular to the separatrix, but stretched in the direction along it. As a result, the real probability of collision is much lower than our original predictions using Λ.



(a) *minimum value of* Λ ***(b)*** *maximum value of* Λ ***(c)*** $\lambda_{min} \times \lambda_{max}$

Figure 4.5: *In the two dimensional flow in equation 4.1, regions of compression in one direction, coincide with expansion in the perpendicular direction. We can see this by plotting both values of* Λ *over a field of points. Regions of compression are indicated by negative values of* Λ *(a), and expansion by positive values of* Λ *(b). Calculating the product of the eigenvalues of* $\mathbf{M}$ *reveals the regions in which volumes surrounding trajectories have shrunk. Simulations were run backwards in time from* 10000 *starting positions using the active tracer. The simulation was run for* 0.5 *characteristic timescales.*

We now have a method of predicting when a specific trajectory will collide with its neighbour given an initial distribution of particles. Crucially, our approach negates the need for pairwise collision detection. Further, by assuming that compression around one trajectory gives an idea about how a number of trajectories behave, we are able to make even greater computational savings. Provided we have a good spread of initial conditions, we are able to approximate collision rates for much larger numbers of particles than are simulated. Each trajectory around which the compression is calculated can represent a family of trajectories as in figure 4.4(b). When the distance between trajectories within the family satisfies equation 4.24, all particles in the family collide. Once collided, the particles are removed from the simulation.

4.2.3 Collisions using the transfer operator

In the previous chapter we introduced a probabilistic approach for modelling the dynamics of inertial particles using a transfer operator. The transfer operator acted on the probability distribution of particle positions in an Eulerian sense and provided local information about particle concentration everywhere in space. Here, we show how knowledge of the local concentration of particles can be used to predict collision rates between particles.

Collision prediction using the transfer operator is bound to a discrete series of integration

points in space. Assumptions about concentrations at the sub-grid scale must be made. The most obvious assumption is that within cells the particle concentration is uniform. Given this assumption, we are interested in the probability that a particle chosen at random is closer to its nearest neighbour than one particle diameter. This information is contained within the cumulative distribution function (CDF) of pairwise distances between particles. The concentration of particles varies in space: the probability of collision will be evaluated cell-wise to provide an expected number of collisions for each cell in our discretised space. As such, we require the CDF of pairwise distances between particles in a regular cube.

Surprisingly, it turns out that the probability distribution of pairwise distances between particles in a box, is both complex and the subject of recent research. The motivation of much of the recent research has been for application in network theory where the distances between nodes in higher dimensional spaces is relevant. For our problem, we desire the probability distribution for pairwise Euclidian distances between particles in 2 and 3 dimensional spaces. Assuming that particles are initially uniformly distributed, Johan Philip [5] provides this for cuboids with arbitrary sides of length a, b and c. The distribution is a function of $y = \|\mathbf{p}_A - \mathbf{p}_B\|^2$, where $\mathbf{p}_A$ and $\mathbf{p}_B$ are the positions of two particles selected at random from within a cell. The resulting expression is long and presented piecewise: broken where the random variable y exceeds various combinations of the three side lengths a, b and c. In our problem, the cells will always be regular squares or cubes, simplifying the expression for the PDF significantly. Through use of the algebraic manipulation program Maple, it was shown in [5] that the PDF of pairwise distances between particles in a 2 dimensional space was given by,

$$f(y) = \frac{1}{b^4}\begin{cases} y + \pi b^2 - 4bb\sqrt{y}, & 0 < y \le b^2 \\ 4b\sqrt{y-b^2} - y + b^2\left[4\arcsin\left(\frac{b}{\sqrt{y}}\right) - 2 - \pi\right], & b^2 < y \le 2b^2 \end{cases} \tag{4.25}$$

In 3 dimensions the expression for the distribution of pairwise distances is significantly more complex,

(4.26)

$$g(z) = \frac{1}{b^6}\begin{cases} 4bz^{\frac{3}{2}} - z^2 + 2\pi b^3\sqrt{z} - 3\pi b^2 z & 0 < z \le b^2 \\ b^3(b(1-3\pi) - 4\pi\sqrt{z}) - (4b^3 + 8bz)\sqrt{z-b^2} + z^2 + & \\ zb^2\left(3\pi + 4\arccos\left(\frac{b}{\sqrt{z}}\right) - 8\arcsin\left(\frac{b}{\sqrt{z}}\right) + 6.1412\right) & b^2 < z \le 2b^2 \\ -1.5z^2 - b^2 z\pi + \sqrt{z-2b^2}(3b^3\pi - 2b^3 - 0.75bz) & \\ -3b^2 z - b^4(\pi + 2.5) - 4b^2(b^2 + z)\left[\arccos\left(\frac{b}{\sqrt{z-b^2}}\right) - \frac{1}{3}\arcsin\left(\frac{b}{\sqrt{z-b^2}}\right)\right] + & 2b^2 < z \le 3b^2 \\ 8b^3\sqrt{z}\left[\arcsin\left(\frac{\sqrt{2z}}{2\sqrt{z-b^2}}\right) - \arccos\left(\frac{b^2}{z-b^2}\right)\right] & \end{cases}$$

where z is the euclidian distance between two randomly selected points in a three dimensional space, $z = \|\mathbf{p}_A - \mathbf{p}_B\|^2$.

To test these distributions, and our implementation of them, we generate 3600 random particle locations along a unit line, within a unit square and within a unit cube respectively. The

positions are taken from a uniform distribution sampled using the MATLAB default Mersenne twister algorithm seeded with the current time. The pairwise distances between the particles are then calculated giving $N_d = N^2/2 - N/2$ distances to analyse, where N_d is the number of unique pairwise distances and N is the initial number of randomly generated positions. The expression for N_d is arrived at because the distance between each pair is only counted once, and the distance between a particle and itself is not required. A histogram H of the pairwise distances is constructed and normalised such that its total area is equal to 1. The normalistation is required so that the histogram is comparable with a PDF. Figure 4.6 shows $H(x)$, $H(y)$ and $H(z)$ for the pairwise distances in 1, 2 and 3 dimensions respectively. The average pairwise distance between the particles increases with the number of dimensions.

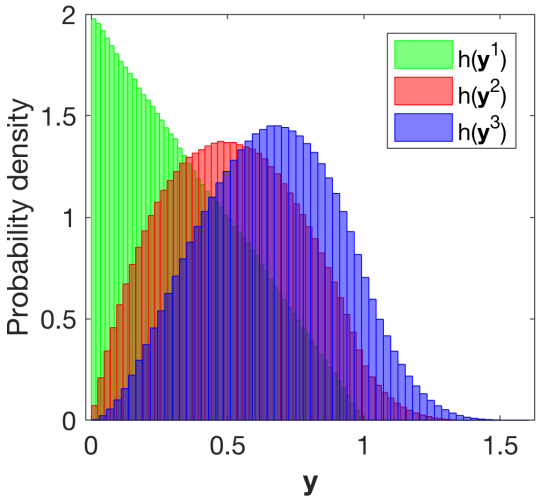
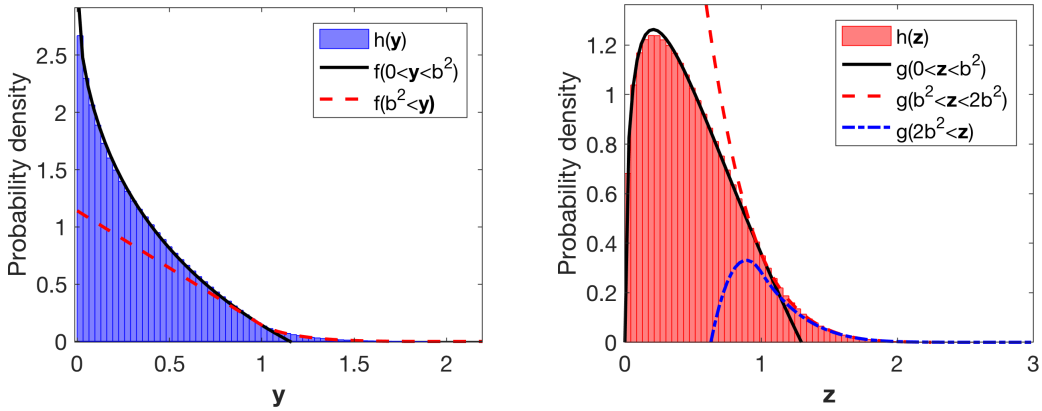


Figure 4.6: *To get an estimate for the distribution of pairwise distances between particles, we generated* 3600 *points from a uniform distribution over a unit line, a unit square and a unit cube respectively. The random variable* y *represents the pairwise distances between each of these points. Each distribution is the discrete probability density function of* y *for* 1, 2 *and* 3 *dimensions. The distributions can be used to find the probability of collisions between particles.*

The analytic expressions in equations 4.25 and 4.26 are defined for the square of the pairwise Euclidian distances. The histograms are therefore recalculated based on the square of the pairwise distances and replotted in figure 4.7. The piecewise distributions calculated in [5] are plotted over the top. The agreement is obvious and excellent.

To calculate the probability of a collision from the PDFs in equations 4.25 and 4.26, we require their corresponding CDFs. The probability of a collision is found by evaluated the CDF at $r_A + r_B$, where r_A and r_B are the radii of two particles.

To find the required CDFs we calculate the integrals of equations 4.25 and 4.26 to give $F(y) = \int_{-\infty}^{\infty} f(y)dy$ and $G(z) = \int_{-\infty}^{\infty} g(z)dz$ respectively. We calculate $F(y)$ and $G(z)$ using help from computer algebra. A result can only be attained for $0 < y \leq b^2$ and $0 < z \leq b^2$. Fortunately,



(a) *Two dimensions* ***(b)*** *Three dimensions*

Figure 4.7: *A comparison can be made between analytically obtained PDFs (red and black lines)[5] and discrete PDFs generated numerically (filled distributions). The distributions match very closely.*

in our simulations, the particle diameter $d_p << b$. Therefore, a CDF defined over the interval $0 < y \leq b^2$ is sufficient.

The calculated CDF for two dimensions is given by,

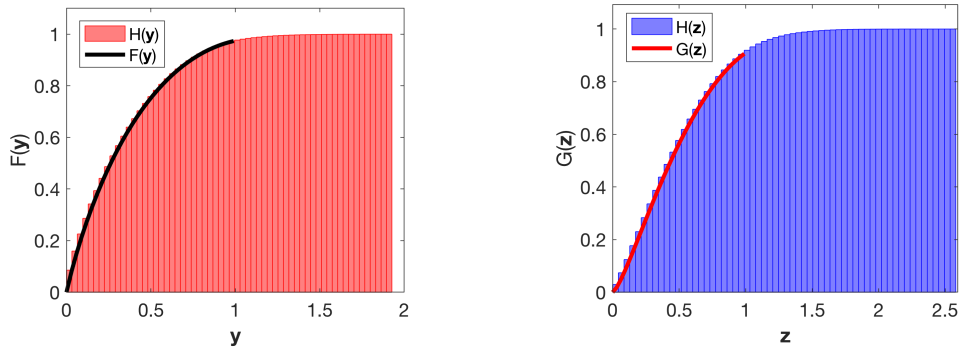
$$F(y) = \frac{3y^2 - 16by^{\frac{3}{2}} + 6\pi b^2 y}{6b^4}, \quad 0 < y \leq b^2 \tag{4.27}$$

and for three dimensions,

$$G(z) = \frac{8z^{\frac{5}{2}}}{5b^5} - \frac{z^3}{6b^6} - \frac{3\pi z^2}{2b^4} + \frac{4\pi z^{\frac{3}{2}}}{3b^3}, \quad 0 < z \leq b^2, \tag{4.28}$$

where as before b is equal to the length of one side of the box. Figure 4.8 shows how the analytical expression for the CDFs, $F(y)$ and $G(z)$, compare with the cumulative sum of the discrete distributions $H(y)$ and $H(z)$ respectively. As discussed, the analytical expression was only successfully obtained for $z \leq b$. Our analytical distributions match exactly the discrete distributions. From the CDFs, the instantaneous probability of a collision between particles of diameter d_p can be calculated. In two dimensions, the probability of a collision, $P(c)$, is given by $F(d_p^2)$, and in three dimensions by, $G(d_p^2)$. We make the assumption that collision events are independent, and can only occur in a pairwise sense (i.e. not in groups of three or four etc.). Therefore, the expected number of collisions, Φ, is given by $\Phi = N_d F(d_p^2)$, where N_d is the number of possible particle pairs.

We recall that the PDF of particle positions predicted by our discretisation of the transfer operator is P_j^n, where n is the current time step and j indicates the cell index. The expected number of particles in the j_{th} cell, N_j, is found by multiplying $\mathbf{P}_j^n$ by the total number of particles in the system, N_p, so that $N_j = \mathbf{P}_j^n N_p$. With a local expected number of particles, the expected



(a) *Two dimensions* ***(b)*** *Three dimensions*

Figure 4.8: *The cumulative frequency distributions for the pairwise Euclidian distances between particles in two and three dimensions have been compared with numerically obtained distributions. Again, good agreement is achieved between analytical expressions and randomly generated points.*

number of collisions is given by,

$$\Phi_j^n = \frac{N_j^2 F(d_p^2)}{2} \tag{4.29}$$

where we have approximated the number of unique collision pairs as, $(N_i^2)/2$. To get an expression for the number of collisions per time step, we multiply Φ_j^n by $\frac{U\Delta t}{\Delta x}$, which is the average number of new configurations of particles passing a cell in one time step.

4.3 Results

4.3.1 Comparing the methods

To compare the methods, we use the simple two dimensional flow field $\mathbf{u}(\mathbf{x})$, given by,

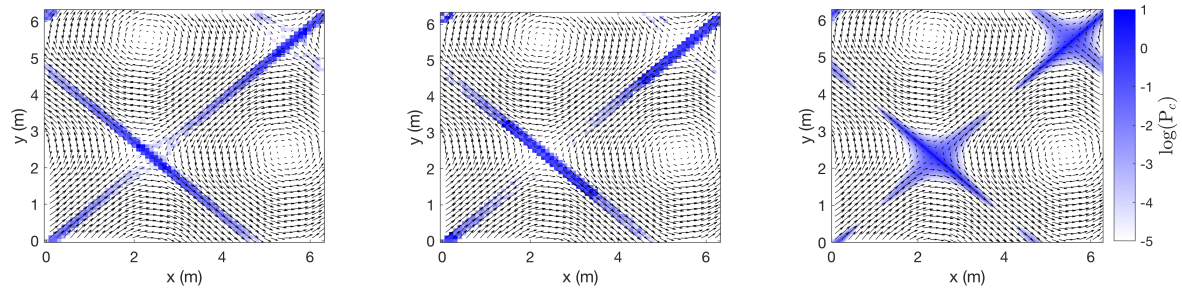
$$\mathbf{u}(x) = \begin{bmatrix} A\cos(y) + B\sin(y) \\ B\cos(x) + A\sin(x)) \end{bmatrix} \tag{4.30}$$

where A and B are flow constants set to $A = 1$ and $B = \sqrt{2}$. A two dimensional flow is chosen because collisions occur readily in a 2-dimensional space. In three dimensions, many more particles are required to produce the same number of collisions.

Within the flow field we simulate $N_p = 5 \times 10^4$ particles with initial positions drawn from a uniform distribution. Where the cell method is used, this translates to 5×10^4 particle trajectories; collisions are detected between the particles once every time step. Where the transfer operator approach is used the densities are initially constant across configuration space and set in each cell to $5 \times 10^4/(nx \times ny)$ where nx ad ny are the number of cells in the x and y directions respectively. Where the eigenvalues of the Cauchy-Green strain tensor are used, we track the compression around 2.5×10^4 individual trajectories: each trajectory is representative of 2 particles. In all

cases the time step Δt is set to 10^{-3}. For the cell method and transfer operator approaches, collisions are assumed to occur in a pairwise manner so that a collision always involves exactly two particles. When a collision occurs both of the particles involved in the collision are removed from the simulation. In the case of the the method using the eigenvalues of the Cauchy-Green strain tensor, all particles within the collided trajectory are removed. The simulations continue for $7.24T$, where the characteristic timescale is given by $T = L/U$. The characteristic velocity U is the root mean square velocity, and L is chosen as the domain size, $L = 2\pi$. We will compare our three different methods based on the predicted collision times, the predicted collision locations, and the algorithmic complexity of the methods.

First, we will investigate the collision locations of the particles. In figure 4.9, the location of collision events are plotted for each of the three methods. The strain tensor approach and cell method produce very similar collision locations, whilst the transfer operator approach predicts collision regions of a different shape. Predicting collisions based purely on local particle concentration, as we do with the transfer operator method, means that particle velocities are not taken into account. In regions where particles are diverging, but are still abundant, collisions are over predicted by the transfer operator.



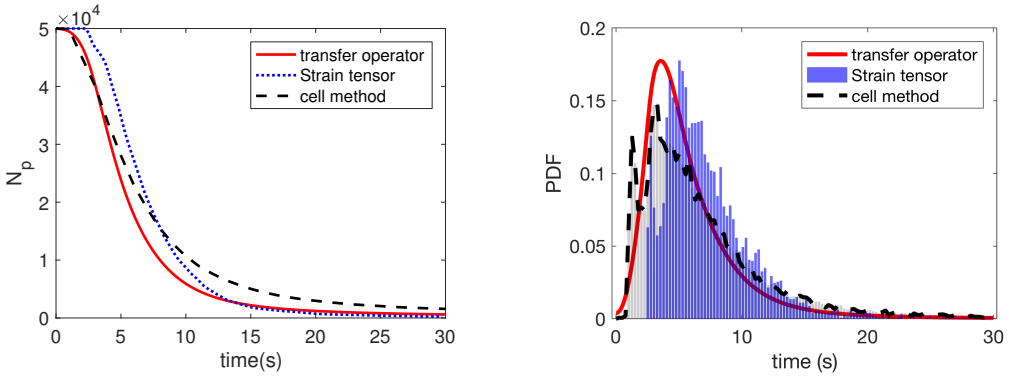
(a) *Cell method* ***(b)*** *Strain tensor approach* ***(c)*** *Transfer operator*

Figure 4.9: *Collisions occur for all three methods where particles cluster on the boundaries between the vortices. Collision regions are shaped subtly differently for each case. The transfer operator predicts collisions based purely on local particle concentration, whereas the other two approaches take into account particle velocities.*

Figures 4.10(a) and (b) show respectively, the number of particles remaining in the simulation, and the probability distribution of collision times. Collisions occur when particles are sufficiently concentrated around a small region either side of the separatrices. In all cases, collision times are governed by the vortex ejection rate. Provided each of the modelling approaches accurately predicts the rate at which particles approach the clustering regions, collision times should be reasonably accurate. For all three cases, good qualitative agreement is achieved although the strain tensor approach slightly under predicts collision rates relative to the cell method. Our strain tensor approach uses the expected initial particle distribution to predict collisions, which results in under prediction of collision rates at the start of the simulation, and over prediction of

collision rates at the end of the simulation. When the cell method is used, particles with nearest neighbours much closer than expected, collide with their neighbours early in the simulation, whereas particles with nearest neighbours further away than expected collide much later in the simulation. Because the strain tensor approach does not take the variation in initial distances into account, the distribution of collision times is narrower, and the curve in figure 4.10(a) steeper.

The transfer operator approach gives slight over prediction of collision rates, but predicts the distribution of collisions over time remarkably well. It should be noted that the success of the transfer operator approach is dependent on sufficiently high collision rates to achieve reasonable comparisons with the other methods. Further, because collisions occur solely along the separatrices, cell sizes must be chosen that produce suitably high particle concentration along the separatix. Improvements to the method are required to extend its validity to more general cases.



(a) *The number of particles remaining in the simulation.*

(b) *Distribution of collision times*

Figure 4.10: *We have used three methods for detecting collisions between inertial particles: the cell method, eigenvalues of the Cauchy-Green strain tensor, and the transfer operator. In (a) the number of particles remaining in the system is cross plotted for each method. Qualitatively similar results are obtained for all three approaches. In (b) we investigate how collisions are distributed in time. We can see that the strain tensor approach slightly under predicts collision rates, whilst the transfer operator produces a reasonably accurate distribution. Particle parameter values are representative of sand in water and given by,* $R = 0.4$*,* $S = 0.0671$*, and* $S/R = 0.1677$*.*

Let us now consider the algorithmic complexity of the methods. As discussed previously, the existing cell method has complexity of $O(N^{\frac{3}{2}})$. Whilst being considerably more efficient than pairwise collision detection across the whole space, the cell method pales in comparison next to the efficiency of the other two methods. When the transfer operator and strain tensor approach are used, the complexity of the simulation is independent of the number of particles represented. With the transfer operator, an unlimited number of particles can be modelled, as the operator is evolving probability densities rather than particle trajectories. The method complexity is instead dependent on the number of cells into which the state space is divided. With the strain tensor approach it is possible to represent more than one particle per simulated trajectory. Again, the

complexity is related to sufficient covering of the particle phase space. The more initial positions evolved using the tangent space equations the better the accuracy of the method.

Despite its algorithmic advantages, the strain tensor approach requires the calculation of the tangent space. The tangent space of the active and history tracers comprises a system of $6d$ equations where d denotes the number of spatial dimensions, which makes trajectories more expensive to simulate. The strain tensor approach also only provides information about the regions through which trajectories have passed. As such, it suffers from issues of statistical convergence. On the other hand, the transfer operator provides a prediction of collision rates throughout the flow in a probabilistic sense. The information about the concentration of the particles is available everywhere and is as valid in regions of low concentration as it is in regions of high.

A weakness of both the strain tensor approach and the transfer operator approach is that they only model collisions between particles of the same diameters. The mass of the particle has a large bearing on the strength of attractors within the flow and will influence collision rates. When attempting to model a process such as raindrop formation, an ability to capture collisions between particles of different sizes is important. After droplets have been growing through various mechanisms for some time, the distribution of droplet masses can be wide, and the flow can contain some significantly larger droplets. This problem can be alleviated by introducing a multi-scale, multi-model approach to modelling raindrop formation, which describes particles of different sizes using different models.

The main advantage of the cell method is that it provides flexibility in both the distribution of particle sizes used, and the types of collisions that can be modelled. When particles collide they can be modelled so that they coalesce or bounce off one another. Models could be devised which incorporate both types of collision in the same simulation depending on the relative velocities of the particles.

In terms of complexity of implementation, the requirement for calculation of the tangent space of particle trajectories makes the strain tensor approach the most difficult. There are approaches to approximate the eigenvalues of the Cauchy-Green strain tensor that are easier to apply to complicated flow fields. By simulating a quartet of trajectories, initially displaced so that they reflect the initial perturbations evolved by the tangent space, an approximation to the eigenvalues of the Cauchy-Green strain tensor can be found. However, over long integration times in chaotic flows, the particles will diverge from one another exponentially, leading to poor approximations to the growth of infinitesimal perturbations offered by the tangent space. The transfer operator provides a straight forward and efficient method for collision modelling. For small size parameter problems with a small range of particle diameters, the transfer operator is the best choice. For larger particles or applications with a wide range of particle diameters the cell method offer the most flexibility.

A clear advantage of both the strain tensor approach and cell method, over using the transfer

operator, is that they can be incorporated into the numerical method detailed in chapter 1. Being able to include history is important in some contexts, as we shall see in the following section.

4.3.2 The influence of the history force on collision rates

It has been shown that including the history force in particle models reduces collision rates in turbulent flows [7]. We argue here that the influence of the history force is due to the reduction of vortex ejection rates. Particles cluster and predominantly collide between the vortices of a flow. If the history force acts to reduce the vortex ejection rates of the particles, then the degree of clustering will be reduced, and hence the number of collisions.

Daitche found that the history force reduced collision rates differently for different density ratios. Here, we investigate two different parameter regimes: a large density ratio example, $\rho_p/\rho_f = 1000$, as is relevant to cloud micro-physics, and a lower density ratio example, $\rho_p/\rho_f = 2$, corresponding to sand in water. To make a meaningful comparison, we fix $St = S/(RT)$ to a constant value of 0.167688, whilst changing the density parameter R. The size parameter must be changed to maintain the value of St. Whilst this can be achieved by changing the particle diameter, the particle diameter has a separate bearing on collision rates. In our simulations, we keep the particle diameter fixed at $5 \times 10^{-4} m$, whilst varying the dynamic viscosity of the fluid, μ. The fluid density, ρ_f, is fixed at 1.

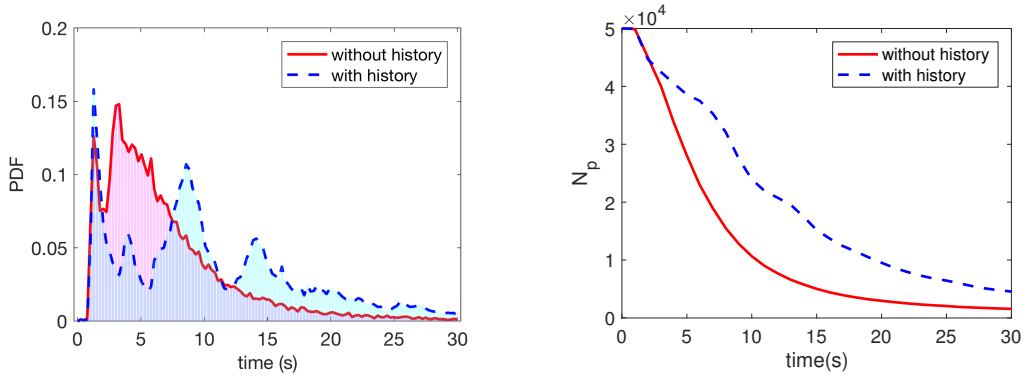
In figure 4.11, we see that particle collision rates are significantly reduced by the inclusion of the history force when density ratios are small. However, when the density ratio is larger, history has little bearing on the collision rates. This result agrees with work in the literature where history was found to reduce collision rates for low density ratios, but increase it for larger ones [7].

A simple 2-dimensional flow gives a clear understanding of why clustering and collision rates are reduced by the history force. When history is included, vortex ejection rates are reduced; particles take longer to reach collision regions between the vortices.

What is more difficult to explain is why history influences the shape of collision regions. In figure 4.13, collision locations are plotted with history (blue circles) and without (red circles). For the case of sand in water in figure 4.13(b) history causes collision regions to spread out in a wide region around the separatrices. When the density ratio reflects water in air, history has little bearing on the locations of collisions.

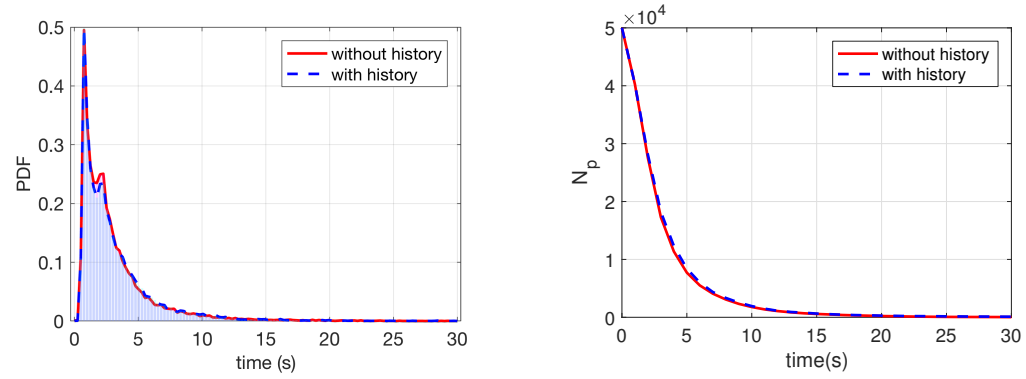
In figure 4.14 we compare the collision regions predicted using the strain tensor approach. As with the cell method, the history force has little bearing on the collision regions for the large density ratio case shown in figure 4.14(a). However, for sand in water, shown in figure 4.14(b), the strain tensor approach exaggerate the influence of the history force. Much wider collision regions appear.

For both modelling approaches, collision regions are larger with the history force included. It appears that the history force can cause trajectories to move closer together in regions of the



(a) *PDF of collision times* ***(b)*** *particles in the system*

Figure 4.11: *Collision times and number of particles remaining in the system are shown with and without history. Fluid and particle densities were representative of sand in water,* $R = 0.4$*. History significantly reduces particle collision rates for this low density ratio case. Particle parameters were given by* $S = 0.1676$*,* $S/R = 0.4190$*,* $St = 0.1011$ *and particle diameter equal to* $5 \times 10^{-4} m$*.*



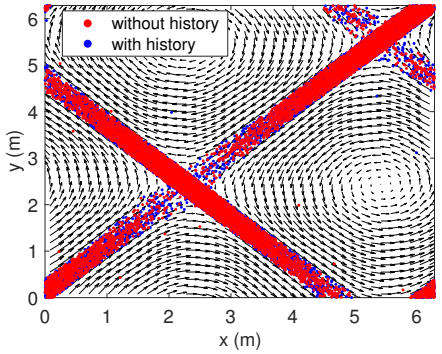
(a) *PDF of collision times* ***(b)*** *No. particles in the system*

Figure 4.12: *For a larger density ratio example, collision times and number of particles remaining in the system are unaffected by the inclusion of history. Simulation parameters were representative of water in air with* $R = 0.001$*,* $S = 0.000419$*,* $S/R = 0.4190$*, and particle diameter equal to* $5 \times 10^{-4} m$

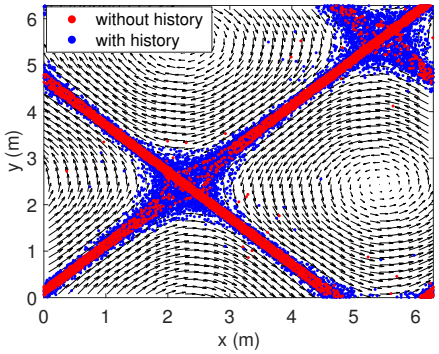
flow that would not necessarily be expected to be collision regions. Because the history force is dependent on long term particle acceleration differences with the fluid, local fluid accelerations are relatively less important than they are for the active tracer. Two history tracers that reach a similar region in the flow, but come from different initial positions, will react differently to the local fluid dynamics because they have a different history of fluid interactions.

4.4 Conclusions

We have compared three collision detection algorithms. One is an existing method from the literature, the cell method, and two are new approaches: the first, which we refer to as the

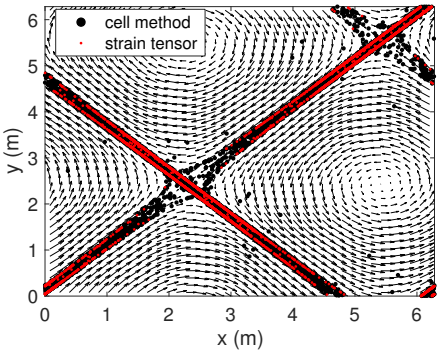


(a) *collision locations for water in air.*

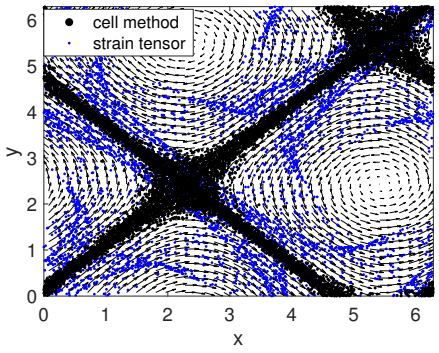


(b) *collision locations for sand in water.*

Figure 4.13: *Collision locations are shown for a simulations lasting for* 7.2 *large eddy turnover times. Collision regions are clearly influenced by the choice to include the history force in modelling for certain parameter regimes. In (a) parameters are representative of water in air corresponding to raindrop formation,* $R = 0.001$*. In (b) parameters were representative of sand in water,* $R = 0.4$*. In both cases* $S/R = 0.4190$*. The particle diameter was set to* $5 \times 10^{-4} m$ *in both cases. Including history in the low density ratio case significantly spreads the collision regions.*



(a) *without history*



(b) *with history*

Figure 4.14: *Collision location comparison between models utilising the strain tensor approach and the cell method reveal that collision regions are influenced by history for the case of sand in water. When history is included the strain tensor approach overestimates the size of the collision region and over predict the collision rate. When history is not included the strain tensor approach predicts the collision locations accurately. Particle parameters were given by* $R = 0.4$ *(representative of sand in water),* $S = 0.1676$*,* $S/R = 0.4190$

strain tensor approach, makes use of the eigenvalues of the Cauchy-Green strain tensor to track compression around particle trajectories; the second uses our discretisation of the transfer operator derived in the previous chapter. The transfer operator approach models collisions based on local particle concentrations, whilst the strain tensor approach predicts collisions based on compression of infinitesimal volumes surrounding trajectories. Both approaches compare well with the cell method in terms of collision locations, and for the example shown, in terms of

collision rates. However, the accuracy of predictions of collision rates using the transfer operator is reduced for lower collision rates, and is affected by the grid resolution. Further work is needed to extend its validity to more general cases, although its potential as a philosophy is clear.

In terms of algorithmic complexity, the transfer operator approach is by far the most successful, as it is able to model collisions between any number of particles independent of the number of particles in the system.

However, both our novel approaches are less flexible than simulating Lagrangian particles. Where the distribution of particle sizes is wide, the cell method is most appropriate. Where particles are small, and similar in diameter, for example in the early stages of raindrop formation, the transfer operator approach is effective.

The history force is also shown to reduce collision rates and change the shape of collision regions when density ratios are small. For larger density ratios, models that do not include history are safe to use.

CHAPTER 5

APPLYING OUR METHODS IN TURBULENT FLOWS

5.1 Introduction

We have developed a transfer operator method that is capable of capturing the preferential concentration of inertial particles in steady flows. However, applications of interest often involve more complicated flows. In applications such as sewage pipe outflow, sedimentation, or raindrop formation, the fluid flow field is a complex picture of turbulent vortical structures. These vortices eject aerosol particles and cause them to preferentially cluster in regions of low vorticity and high strain rate [18]. In regions of high particle concentration the interaction between particles is increased, and the interaction between the particles and the fluid is also affected. Small scale velocity fluctuations are excited by the particles changing the form of the turbulent energy spectrum [20]. A method capable of capturing the inertial behaviour of the particles in the continuum limit, allows these phenomena to be investigated efficiently.

Appropriate choice of particle model is a concern for turbulent flows, as it has been throughout the document. Specifically, questions arise over when it is necessary to include the Basset history force. Despite there being many studies that investigate the influence of the history force in complicated turbulent flows, the subject remains an open research area. An early example of work on the simulation of inertial particles in turbulence was conducted by Elghobashi and Truesdell [92], at a time when computing resources were scarce. Elghobashi and Truesdell investigated the behaviour of inertial particles in homogenous and isotropic turbulence simulated over a coarse grid. They investigated the relative importance of each of the forces acting on the particles and found that in general the Basset force was approximately 1/10 of the the magnitude of the drag term. When gravity was included, the buoyancy force was the most important force. However, a relatively small parameter sweep was conducted. For each density ratio, only one

particle diameter was investigated. Armenio and Fiorotto [93] investigated the relevance of the forces acting on inertial particles in a turbulent channel flow, and found the Basset force to be significant for all parameter values. In neither case was the clustering behaviours of the particles investigated.

Recently, the interest in the area has been reignited and a number of studies have been concerned with the influence of the history force on the ensemble behaviour of inertial particles [7, 31, 72, 73]. Anton Daiche investigated the influence of history for a range of density ratios and Stokes numbers [7]. Daitche found that the history force significantly reduces clustering and collisions for small density ratios, but had a lesser effect for large density ratios such as in raindrop formation. Olivieri also found that the history force reduced preferential concentration, but found that history played a role even in the larger density ratio case of raindrop formation [31]. Both Daitche and Olivieri neglected gravity in their simulations. The influence of history in the presence of gravity was investigated by Guseva et al. [72]. A chaotic flow with well studied attractors was used as a paradigmatic example. The behaviour of the particles with respect to the attractors in the flow had been studied in the absence of the Basset force. Guseva showed how the Basset force changed the nature and number of attractors in the flow when history was included. Further, she showed that vertical trapping of particles occurred in the absence of history, but leaked from the attractors when history was included. In all cases, the influence of history is found to increase for larger particle diameters. For further examples of studies concerned with the preferential concentration and dispersion of inertial particles in turbulent flows, see [94] for a recent review.

In this chapter, we investigate the efficacy of the transfer operator approach in turbulent flows. We compare the results of the preferential concentration predicted by the transfer operator with the clustering of simulations of Lagrangian particles. For our fluid flow, we choose the turbulent breakdown of the ABC flow. For the particle parameters we investigate two cases representative of interesting natural world phenomena. The first is the large density ratio case of raindrops in air, where $\rho_p/\rho_f \approx 1000$. The second is analogous to the sedimentation of sand in water, where $\rho_p/\rho_f \approx 2$. Armed with an understanding of how the method behaves in more realistic flows, we will be able to apply it to our original motivating modelling problems, sedimentation, and raindrop formation.

We also investigate the influence of the Basset history force on the clustering of inertial particles. We compare our results with the recent studies of Daitche, Olivieri and Guseva [7, 31, 72]. Because our transfer operator approach is designed to capture the behaviour of particles without history, it is important for us to understand how the history force influences inertial particles in turbulent flows. With said knowledge, applications in which the transfer operator is most useful can be identified.

First, the integration of the transfer operator with a numerical scheme for the solution of the Navier-Stokes equations is discussed. We then show how the particle parameters can be extracted

from a turbulent energy spectrum. Our transfer operator approach is then compared with simulations of large numbers of Lagrangian particles, in a flow initialised with a superposition of ABC flows. Finally we use the same flows to investigate the influence of the Basset history force on the preferential clustering of the particles.

5.2 Numerical Implementation

5.2.1 The fluid phase

Our transfer operator method is integrated with a numerical solver of the Navier-Stokes equations. We use a finite volume, incompressible flow solver called MOBILE, which is designed to handle multiple miscible fluids. To deal with multiple fluids of different density, the Boussinesq approximation is made. Starting from the compressible form of the Navier-Stokes equations, under the Boussinesq approximation, the fluid density is assumed constant in all terms except the buoyancy term. The approximation leads to the incompressible form of the Navier-Stokes equation with the addition of a variable density buoyancy term, ρg_i. The equations are given by,

$$\frac{\partial u_i}{\partial x_i} = 0 \quad i = 1,2,3 \qquad a)$$

$$\rho_f \frac{\partial (u_i)}{\partial t} + \rho_f u_j \frac{\partial (u_i)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \rho g_i + \mu \frac{\partial^2 u_i}{\partial x_j \partial x_j} \qquad b) \tag{5.1}$$

where u_i is the fluid velocity in the i^{th} dimension, ρ_f is the constant density of the fluid, p is the fluid pressure, μ is the dynamic viscosity of the fluid, g_i is gravity and ρ is the variable component of the density in the buoyancy term.

Studying equation 5.1 we can see that we have two unknowns and only one equation containing both. As such we have no method for updating the pressure. To produce an equation for the pressure, the divergence of the momentum equation, 5.1(b), is taken to give,

$$\rho_f \frac{\partial}{\partial x_i}\left(\frac{\partial u_i}{\partial t}\right) + \rho_f \frac{\partial}{\partial x_i}\left(u_j \frac{\partial u_i}{\partial x_j}\right) = -\frac{\partial}{\partial x_i}\left(\frac{\partial p}{\partial x_i}\right) + \rho \frac{\partial g_i}{\partial x_i} + \mu \frac{\partial}{\partial x_i}\left(\frac{\partial^2 u_i}{\partial x_j \partial x_j}\right). \tag{5.2}$$

Under the Bousinesq approximation, derivatives of ρ are equal to zero. Through the product rule, and using the fact that $\partial/\partial x_i$, $\partial/\partial x_j$ and $\partial/\partial t$ commute, we reach,

$$\rho_f \frac{\partial}{\partial t}\left(\frac{\partial u_i}{\partial x_i}\right) + \rho_f \frac{\partial u_j}{\partial x_i}\frac{\partial u_i}{\partial x_j} + u_j \frac{\partial}{\partial x_j}\left(\frac{\partial u_i}{\partial x_i}\right) = -\frac{\partial^2 p}{\partial x_i \partial x_i} + \mu \frac{\partial^2}{\partial x_j \partial x_j}\left(\frac{\partial u_i}{\partial x_i}\right) \tag{5.3}$$

and applying the condition of incompressibility, we arrive at,

$$\rho_f \frac{\partial u_i}{\partial x_j}\frac{\partial u_j}{\partial x_i} = -\frac{\partial^2 p}{\partial x_i \partial x_i}, \quad (a)$$
$$\rho_f \frac{\partial u_i}{\partial t} + \rho_f u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \rho g_i + \mu \frac{\partial^2 u_i}{\partial x_j \partial x_j}. \quad (b) \tag{5.4}$$

A numerical solution of equations 5.4(a) and 5.4(b) provides a method for updating the pressure and velocity whilst incorporating the condition of incompressibility implicitly through 5.4(a) [95].

MOBILE follows the implicit large eddy simulation (ILES) approach to solving the Navier-Stokes equations. In ILES approaches the viscous term from the Navier-Stokes equations, $\mu \frac{\partial^2 u_i}{\partial x_j \partial x_j}$, is neglected. The viscosity forcing is instead incorporated implicitly through numerical diffusion and we are left with,

$$\rho_f \frac{\partial u_i}{\partial x_j}\frac{\partial u_j}{\partial x_i} = -\frac{\partial p}{\partial x_i \partial x_i}, \quad (a)$$
$$\rho_f \frac{\partial u_i}{\partial t} + \rho_f u_j \frac{\partial (u_i)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \rho g_i, \quad (b) \tag{5.5}$$

Equation 5.5 is solved numerically using a third order, finite volume, monotonicity preserving upwind scheme [95]. For further specific details on the numerical solution of the Navier-Stokes used in this work see [95], and for studies verifying the numerical scheme see for example [96].

5.2.2 The particle phase

We now discuss the integration of the particle phase with the Navier-Stokes solver MOBILE. Both the transfer operator method detailed in the third chapter, and the numerical solution of the MR equation introduced in the second chapter, have been linked with MOBILE. The program flow can be described as follows: the fluid phase is updated each time step using MOBILE. For each new value of the fluid velocity, the transfer operator is recalculated using the method detailed in the third chapter. Specifically, particles are initialised on corners of each grid cell with initial velocities predicted by the inertial equation. The particles are then advected for a short time and their new positions define a shape with a new centroid. Each cell is then redistributed over its neighbouring cells based on the translation of the cell centroid.

The three step update process, (fluid -> particle -> remapping), requires three discrete time parameters. The first of these is the time step used to solve the fluid phase, Δt_f. The fluid time-step is limited to ensure that parcels of fluid are not transported by more than one cell in a time step. This condition is known as the Courant-Friedrichs-Lewy (CFL) condition and ensures stability of the scheme [97]. The CFL condition is given by $\Delta t \sum_i \frac{u_i}{\Delta x_i} \leq C_{max}$, where Δx_i is the cell spacing, u_i is the fluid velocity in the i^{th} direction, and C_{max} is some positive constant typically equal to 1.

The second discrete time parameter, Δt_p, is the time period over which particles are advected before remapping. We will refer to Δt_p as the operator time step. Recall that the transfer operator is a linear operator W_{ij}, which acts on the discretised PDF of particle positions Q_j^n, such that,

$$Q_i^{n+1} = W_{ij} Q_j^n \tag{5.6}$$

where superscript n denotes the current time step. The operator time step, therefore, corresponds to one multiplication of the PDF of particle positions by the transfer operator. Or, in other words, the operator time step corresponds to the length of time between time steps n and $n+1$.

In general, Δt_p is set equal to Δt_f. However, inertial particles do not follow fluid streamlines exactly. In some instances it is possible that particles are displaced by more than one cell in within a fluid time step. To prevent particles from travelling by more than one cell per time step, the particle phase includes its own CFL condition. Conditions on the time step must be enforced based on the maximum particle velocity in the domain. The maximum particle velocity in the domain, $\max\{\mathbf{v}(\mathbf{x})\}$, must satisfy $||\max\{\mathbf{v}(\mathbf{x})\}\Delta t_p|| < 0.75\Delta x$, where $\mathbf{x}$ is a position in the domain and Δx is the grid resolution. If the particle CFL condition is violated, Δt_p is set to the highest possible factor of Δt_f that satisfies the condition.

The third discrete time parameter is the time step used for the particle trajectory integration, Δt. We will refer to Δt as the particle time step. The particle timestep is the time step used for the numerical method detailed in the second chapter and is set for all time to $\Delta t_p/5$.

The transfer operator method is defined over a regular cartesian grid in space. In a numerical simulation over a regular grid, the fluid velocity is only available at discrete integration points in space and time. To attain the continuous velocity field required by the Maxey-Riley equation, we use trilinear interpolation to determine the within cell fluid velocities. To find the fluid velocity between fluid time steps, the fluid velocity field is interpolated linearly in time. By using both interpolation schemes together, we are able to calculate the velocity required by the particle phase. Let us consider that the point $\mathbf{x} = (x, y, z)$ is within a cell with vertex positions given by $\mathbf{v}_j = (x_j, y_j, z_j)$. With knowledge of the fluid velocity at the vertices, $\mathbf{v}_j$, the fluid velocity at time t, and point $\mathbf{x} = (x, y, z)$, is given by,

$$u_i(\mathbf{x}, t + n\Delta t) = \sum_{j=0}^{7} N_j \left[u_i(\mathbf{v}_j, t + n\Delta t) + \left(\frac{n\Delta t}{\Delta t_f} \right) \delta u(\mathbf{v}_j, t) \right], \tag{5.7}$$

where n is the current sub time step, $n \in [0, 1, 2, 3, 4]$, $\delta u(\mathbf{v}_j, t)$ is the velocity difference between time steps given by $\delta u(\mathbf{v}_j, t) = u(\mathbf{v}_j, t) - u(\mathbf{v}_j, t - \Delta t_f)$ and N_j is a vector of weights. For a particle

at position $\mathbf{x}$, the vector of weights, N_j, is given by,

$$
\begin{aligned}
N_0 &= \frac{(x_1-x)(y_1-y)(z-z_0)}{\delta V}, \; N_1 = \frac{(x_1-x)(y-y_0)(z-z_0)}{\delta V}, \\
N_2 &= \frac{(x-x_0)(y_1-y)(z-z_0)}{\delta V}, \; N_3 = \frac{(x-x_0)(y-y_0)(z-z_0)}{\delta V}, \\
N_4 &= \frac{(x_1-x)(y_1-y)(z_1-z)}{\delta V}, \; N_5 = \frac{(x_1-x)(y-y_0)(z_1-z)}{\delta V}, \\
N_6 &= \frac{(x-x_0)(y_1-y)(z_1-z)}{\delta V}, \; N_7 = \frac{(x-x_0)(y-y_0)(z_1-z)}{\delta V},
\end{aligned} \tag{5.8}
$$

where $\delta V = \Delta x \Delta y \Delta z$, $\mathbf{x_0} = (x_0, y_0, z_0)$ and $\mathbf{x_1} = (x_1, y_1, z_1)$ are positions in space and Δx, Δy, Δz are the grid resolutions in each respective dimension [98]. Each of the vertex positions can be seen in figure 5.1.

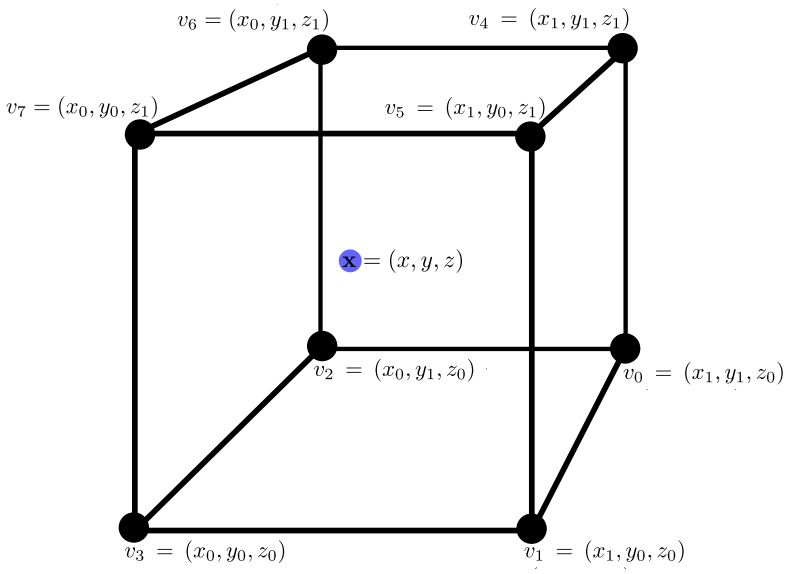


Figure 5.1: *The velocity of the fluid between integration points is found through trilinear interpolation. Black points in the figure mark positions of the vertices used for trilinear interpolation. The velocity at the marked point x is a linear combination of the velocities at the vertices v_0 to v_7, which is found using equations (5.7) and (5.8).*

For further information on the code produced to model the particle phase, a user guide has been presented in the appendices.

5.2.3 Characterising the fluid velocity field

Modelling inertial particles requires certain knowledge about the flow in which they reside. Here, we discuss how best to characterise the flow and how to extract the parameters relevant to the particles. Characteristic time and length scales are well defined for steady flows. Often, the

diameter of flow features is easy to extract and the time over which they evolve calculable. In turbulent flows however, a range of length and time scales exist. The length scales range from the Kolmogorov length scale η [26], to the scales of the largest eddies. In the cases of cloud flows, the largest scales can be of the order of kilometres whilst η is of the order of millimetres. The Kolmogorov scale is the scale at which viscous forces dominate and the energy in the flow is dissipated into heat. Small inertial particles interact with the smallest scales in the flow and the parameters of the particles must be defined accordingly.

The Stokes number St, previously defined as $\frac{2r_p^2 Re}{9RL^2}$, is defined differently for turbulent flows such that it reflects the Kolmogorov scales. The Stokes number is the decay rate of the velocity difference $\mathbf{v}(t)-\mathbf{u}(x,t)$, divided by a timescale of the fluid. In the previous chapters, this ratio was found using the large eddy turnover time $T=U/L$, where U is the U_{rms} of the steady flow field, and L is a characteristic length scale. In the previous chapters the largest eddies were of the order of the domain size and so the domain size provided a natural length scale. In turbulent flows, the Stokes number is given by $St=\tau_p/\tau_\eta$ where $\tau_p=\frac{2r_p^2\rho_f}{9R\mu}$ is the particle response time, and τ_η is the Kolmogorov timescale. The Kolmogorov timescale is given by $\tau_\eta=\sqrt{\mu/(\rho_f\epsilon)}=\eta/u_\eta$, where ϵ is the mean energy dissipation of the flow, μ is the dynamic viscosity of the fluid, ρ_f is the fluid material density and u_η is the Kolmogorov velocity scale. The Kolmogorov scale must be used for turbulent flows: it is the timescale over which the smallest flow features evolve, and small inertial particles are influenced by the smallest length scales in the turbulent flow.

Whilst it is important that the flow is resolved close to the Kolmogorov length scale, it is computationally impractical to simulate all length scales. Simulating all scales from smallest to the largest, in a flow with parameter regimes akin to a cloud or a turbulent body of water, requires a simulation able to resolve scales that cover 6 orders of magnitude. Instead, we must make a compromise. By simulating a flow in a cube with periodic boundary conditions, a small domain is able to represent a small section of a cloud-like flow. Because only a small section of the cloud is simulated, we sacrifice the largest scales in the flow in favour of an ability to resolve the smallest.

A Stokes number defined as, $\frac{2r_p^2\rho_f}{9R\mu\tau_\eta}$, is dependent on both the density parameter and the particle diameter. To elucidate the influence of density from that of the particle size, we recall the size parameter, S. The size parameter, $S=\frac{2r_p^2\rho_f}{9\mu}$, is dependent only on the particle diameter and not the density ratio. In our simulations we do not non-dimensionalise the equations of motion during the simulation. Effectively the parameter τ_η is set to 1. Instead, we use the size parameter S and density ratio R throughout.

Using S and R the equations governing the motion of the particle phase are the active tracer equation given by,

$$\dot{\mathbf{v}}(t)=\frac{3}{2}R\frac{D\mathbf{u}}{Dt}-\frac{R}{S}(\mathbf{v}(t)-\mathbf{u}(\mathbf{x},t)) \tag{5.9}$$

and the full MR equation given by,

$$\dot{\mathbf{v}}(t) = \frac{3}{2}R\frac{D\mathbf{u}}{Dt} - \frac{R}{S}(\mathbf{v}(t) - \mathbf{u}(\mathbf{x},t)) - \frac{3R}{\sqrt{2\pi S}}\int_{t_0}^{t}\frac{1}{\sqrt{t-\tau}}\frac{d}{d\tau}(\mathbf{v}(\tau) - \mathbf{u}(\mathbf{x},\tau))d\tau, \tag{5.10}$$

where, $\mathbf{v}(t)$ is the particle velocity, $\mathbf{u}(\mathbf{x},t)$ is the velocity of the fluid and $\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + u \cdot \nabla \mathbf{u}$ is the material derivative.

In our simulations we choose to investigate the behaviour of inertial particles in a decaying turbulent flow. A decaying flow allows investigation into the degree of preferential concentration as the energy in the flow dissipates. In such flows, the effective Stokes number changes throughout the simulation. We show that preferential concentration plotted as a function of S/R varies throughout the simulation, but when S/R is normalised by U, the results collapse onto the same curve. By not including U, L or τ_n in the parameters of our particles during simulation, comparisons can be readily made with the sedimentation rate of, for example, particles settling under gravity in a quiescent fluid where the choice for U and L is unclear.

Despite not requiring the characteristics of the turbulent flow for calculation of our particle parameters, it is important to accurately characterise the turbulent flow field. Knowledge of the defining features of the flow field will allow comparison with other studies. Many of the parameters that classify a turbulent flow can be calculated from the turbulent energy spectrum. The energy density $E_{ij}(\mathbf{k},t)$ of the flow is given by,

$$E_{ij}(\mathbf{k},t) = \frac{1}{2}\widehat{R}_{ij}(\mathbf{k},t), \tag{5.11}$$

where we require $R_{ij}(\mathbf{r})$ and its Fourier transform $\widehat{R}_{ij}(\mathbf{k})$. The function $R_{ij}(\mathbf{r})$ is the autocorrelation of the fluid velocity field, given by,

$$\begin{aligned} R_{ij}(\mathbf{r}) &= \frac{1}{V}\int_V \mathbf{u}(\mathbf{x})\mathbf{u}(\mathbf{x}+\mathbf{r})d\mathbf{x} \\ \widehat{R}_{ij}(\mathbf{k}) &= \int_{-\infty}^{\infty} e^{-2\pi\hat{i}\mathbf{r}}R(\mathbf{r})d\mathbf{r}. \end{aligned} \tag{5.12}$$

where, $\mathbf{k}$ is the wave vector and denotes a position in the Fourier domain, $\hat{i}$ is the $\sqrt{-1}$ and $\mathbf{x}$ and $\mathbf{r}$ are positions in space [26]. By making use of the autocorrelation theorem it is possible to find $R_{ij}(\mathbf{r})$ as a product of two Fourier transforms. The auto correlation theorem states that in a domain with volume V,

$$\frac{(2\pi)^3}{V}\hat{\mathbf{u}}(\mathbf{k})\hat{\mathbf{u}}^*(\mathbf{k}) = \int_{-\infty}^{\infty} e^{-2\pi\hat{i}\mathbf{r}}\left[\frac{1}{V}\int_V \mathbf{u}(\mathbf{x})\mathbf{u}(\mathbf{x}+\mathbf{r})d\mathbf{x}\right]d\mathbf{r}, \tag{5.13}$$

where $\hat{\mathbf{u}}$ is the Fourier transform of $\mathbf{u}(\mathbf{x},t)$ and $\hat{\mathbf{u}}^*$ is the complex conjugate of $\hat{\mathbf{u}}$. To take advantage of equation 5.13 we calculate the 3-dimensional Fourier transform of $\mathbf{u}(\mathbf{x},t)$. The Fourier transform is conducted using $MATLAB$'s `fftn` function allowing the autocorrelation $\widehat{R}_{ij}$ to be calculated as,

$$\begin{aligned} \hat{\mathbf{u}}(\mathbf{k}) &= \text{fftn}(\mathbf{u}(\mathbf{x})) \\ \widehat{R}_{ij}(\mathbf{k}) &= \hat{\mathbf{u}}(\mathbf{k})\hat{\mathbf{u}}(\mathbf{k})^*, \end{aligned} \tag{5.14}$$

where $\mathbf{u}(\mathbf{x},t) = [u(\mathbf{x}), v(\mathbf{x}), w(\mathbf{x})]$. The total energy density is the sum of the contributions from each component of $\mathbf{u} = [u, v, w]$ so that $\widehat{R}_j(\mathbf{k},t) = (\hat{u}\hat{u}^*)_j + (\hat{v}\hat{v}^*)_j + (\hat{w}\hat{w}^*)_j$.

We now have the energy $E_j(\mathbf{k},t)$ in wave number space, as a function of the wave vector $\mathbf{k}$. However, we desire the 1 dimensional energy spectrum given by $E(||\mathbf{k}||,t)$, where $||\mathbf{k}|| = \sqrt{{k_x}^2 + {k_y}^2 + {k_z}^2}$. The contributions of $E_j(\mathbf{k},t)$ to $E(||\mathbf{k}||,t)$ exist in spherical shells of thickness δk. Numerical integration over the spherical shells is conducted to find the total energy as a function of $||\mathbf{k}||$. By finding the average energy contained within each shell and multiplying it by the surface area of the shell, we can find the energy contained within each shell. The surface area of each shell is simply the surface area of a sphere, $4\pi||\mathbf{k}||^2$. Finally, the 1-dimensional energy spectrum is given by $E(||\mathbf{k}||,t) = 2\pi||\mathbf{k}||^2\widehat{R}(||\mathbf{k}||)$. Figure 5.2, inspired by [26], illuminates this process. A shell of thickness δk is highlighted in light blue. The average of the values in the light blue shell is found to give a value of $E(||\mathbf{k}||,t)$.

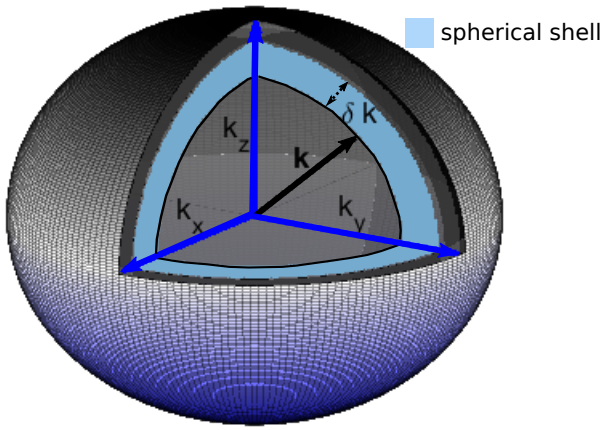


Figure 5.2: *To find the energy density as a function of $||\mathbf{k}||$, the average energy is found within a series of spherical shells of thickness δx. The figure shows a spherical shell in the Fourier domain. The values contained within the blue band are averaged over to give $E(||\mathbf{k}||,t)$.*

Using $E(||\mathbf{k}||,t)$ we define the flow parameters as follows. Following [7], we define a characteristic velocity U, as the root mean square velocity $U = \sqrt{\frac{1}{\mathbf{V}}\int_{\mathbf{V}}|u(\mathbf{x})|^2 d\mathbf{V}}$, where $\mathbf{V}$ is the volume

of the domain. The characteristic length of the flow, L, is defined as the integral length scale, $L = \frac{\sum_k k^{-}1E(k,t)}{\sum_k E(k,t)}$, where k is the spatial frequency or wavenumber, and $E(k,t)$ is the energy spectrum as a function of wavenumber [99]. The integral length scale corresponds to the reciprocal of the centroid of the energy spectrum, $E(||\mathbf{k}||,t)$. If K_c were the centroid of the energy spectrum, then $L = 1/K_c$.

To quantify the decay of energy in the flow, we calculate the mean energy dissipation rate ϵ. The dissipation rate can be calculated following several different approaches. Here we calculate ϵ as $\epsilon = dE(t)/dt \approx E(t+\Delta t) - E(t)/\Delta t$. It has been shown that ϵ can also be calculated as $\epsilon = C_\epsilon U^3/L$ where C_ϵ is some Re independent constant. See [6] and the references therein. We compare both methods whenever ϵ is calculated.

5.2.4 Initial and boundary conditions of the fluid velocity field.

We choose two distinct tests for the transfer operator. Each acts as an analogue for a different natural phenomenon. The first is the decay of turbulent vortices in a periodic cube, which stands in for a cloud like flow. A decaying flow allows us to investigate both the clustering and sedimentation properties of the particles as the turbulent intensity decreases. The second test uses the same turbulent vortices, but includes gravity and represents the sedimentation of sand particles in a river. Such a flow is the perfect analogue for a tidal river basin. In a tidal river basin, the sediment is lifted up and carried when the tidal current is strongest. As the the flow subsides closer to slack tide, the sediment is deposited by the flow.

Following the work of Teitelbaum and Mininni [99] and Rorai et al. [100], we investigate the turbulent breakdown of the ABC flow. Initially, the fluid velocity field is given by a superposition of ABC flows which quickly decay into turbulence. We choose the superposition of ABC flows to allow comparison with previous works. In future work we would like to investigate the influence of the particle phase on the turbulence. A reference point in the literature allows robust comparisons to be made. The initial velocity field, $\mathbf{u}(\mathbf{x},0)$, is given by,

$$\mathbf{u}(\mathbf{x},0) = \sum_{k_0=1}^{14} \begin{bmatrix} C\cos(2\pi k_0 y) + A\sin(2\pi k_0 z) \\ A\cos(2\pi k_0 z) + B\sin(2\pi k_0 x) \\ B\cos(2\pi k_0 x) + C\sin(2\pi k_0 y) \end{bmatrix}, \tag{5.15}$$

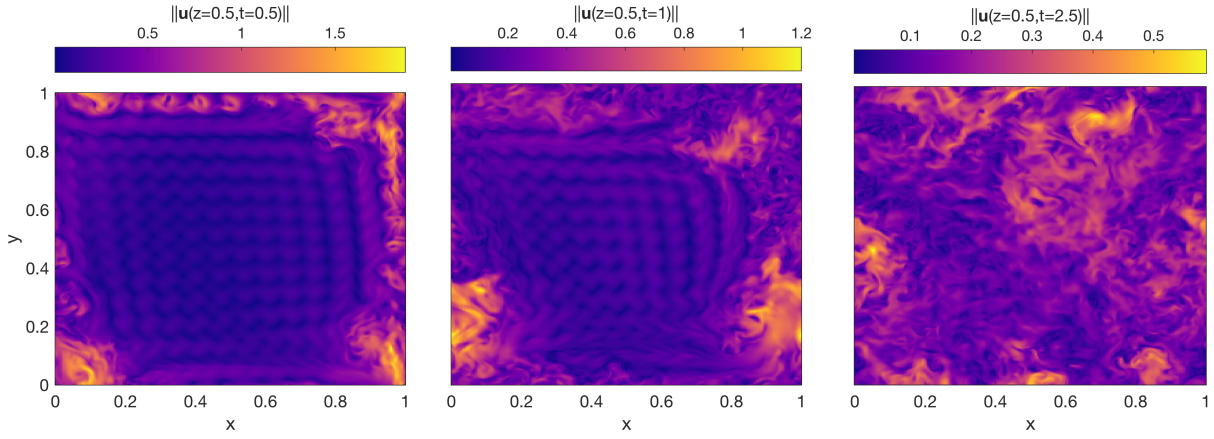
where $A = 0.09$, $B = 0.1$ and $C = 0.11$. A combination of fourteen ABC flows is used to match the work of previous authors.

The flow defined by equation 5.15 contains a range of length scales due to the superposition of ABC flows of differing initial spatial frequency, k_0. The initial structure in the velocity field decays quickly into a homogenous and isotropic turbulent flow. The first few seconds of the decay are shown in figure 5.3 for the case of periodic boundary conditions. Figure 5.3 shows the magnitude of the velocity, $||\mathbf{u}||(\mathbf{x},t)$ in the x-y plane at $z = 0.5$ for three instances in time: $t = 0.5$, $t = 1$ and $t = 2.5$ seconds. Initially, the flow has clear periodic vortices and a root mean square velocity of

A	B	C	Δ_x	η	$domain$	ρ_f	τ_η	U_0	μ
0.09	0.1	0.11	0.00390	$3\Delta x$	1	1000	η/u_η	0.3754	1.003×10^{-3}

Table 5.1: *The model parameters used for both the unbounded (excluding gravity) and bounded simulations (including gravity).*

$U = 0.3754$. After 2.5 seconds the flow has become turbulent and much of the initial structure in the flow is gone. In all simulations the number of cells is $N = 256^3$, the fluid density is set to $\rho_f = 1000$, and the dynamic viscosity is $\mu = 1.003 \times 10^{-3}$.



(a) $t = 0.5$ ***(b)*** $t = 1$ ***(c)*** t=2.5

Figure 5.3: *The magnitude of the velocity field in the x-y plane at $z = 0.5$ is shown for three instances in time. The fluid velocity field was initially given by equation 5.15, but quickly turns to a turbulent flow. The flow constants were given by, $A = 0.09$, $B = 0.1$, $C = 0.11$, and the number of cells is equal to $N = 256^3$.*

In the second example we demonstrate the application of the transfer operator to a sedimentation problem. The flow field is initialised with the same initial conditions, however, as the simulation is designed to represent sedimentation, the boundary conditions are modified. The flow is bounded in the vertical by two walls. At the bed a no-slip boundary is implemented, whilst at the surface the velocity is restricted only in the vertical. For a velocity field $\mathbf{u} = [u, v, w]$, the boundary condition at the bed is given by, $\mathbf{u}([x, y, 0], t) = [0, 0, 0]$ and the boundary condition at the surface is given by $w([x, y, 1], t) = 0$. Again the domain is a 1 metre cube and A, B and C, in equation 5.15, are set to $A = 0.09$, $B = 0.1$, $C = 0.11$.

Tables (5.1) and (5.2) detail the flow parameters used in all simulations in this chapter. The values for the particle phase are chosen so that $St = \frac{S}{R\tau_n}$ ranges from 1×10^{-3} to 100. The largest values of Stokes will certainly cause the particle Reynolds number to exceed a safe value, but are included to show how the Maxey-Riley equation predicts the destruction of clustering for large particles. The parameters in table 5.1 and table 5.2 are used for three distinct modelling cases.

S	R	Representative of:
$\in [10^{-4}, \quad 40]$	0.4	sand in water
$\in [2.5\times10^{-7}, \quad 2\times10^{-2}]$	0.0012	rain droplets in air

Table 5.2: *The range of particle parameters used in all simulations.*

First we model individual particle trajectories described by the active tracer given in equation 5.9. Second we use the same parameter ranges for simulations calculated by the transfer operator method introduced in the third chapter. Finally, we investigate the individual particle trajectories including the history force. We use the numerical method detailed in the second chapter to calculate the particle trajectories.

All three modelling approaches will be compared in the proceeding sections. First, we calculate the energy spectra of the two flows, $E(||\mathbf{k}||,t)$, from which the remaining flow parameters can be calculated. We calculate the energy spectra of the flows using the method detailed in the previous section. The resulting energy spectrum for the bounded flow is shown in figure 5.4 and figure 5.5. Spectra have been plotted for a number of instances in time between 0 and 30 seconds. The inset shows how $E(t) = \int_{-\infty}^{\infty} E(||\mathbf{k}||,t)dk$ evolves with time. Using $E(||\mathbf{k}||,t)$ we can approximate the integral length scale L, the Kolmogorov length scale η, and the mean energy dissipation rate ϵ.

As discussed in earlier sections, small inertial particles are influenced by the smallest length scales in a turbulent flow. The smallest scales in time, length, and velocity are known as Kolmogorov's micro-scales [26]. Given the energy dissipation rate,

$$\epsilon(t) = \frac{dE(k,t)}{dt} \tag{5.16}$$

we can calculate each of Kolmogorov's micro-scales for our flows. Kolmogorov's micro-scales are given by,

$$\begin{aligned} \eta &= \left(\frac{\nu^3}{\epsilon}\right)^{\frac{1}{4}} \\ \tau_n &= \left(\frac{\nu}{\epsilon}\right)^{\frac{1}{2}} \\ u_n &= (\nu\epsilon)^{\frac{1}{4}}. \end{aligned} \tag{5.17}$$

By plotting each of the micro-scales on the same graph as the large scales, L and U, we can get a clear picture of the characteristics of the flow. Figure 5.6 shows the characteristic flow velocity, length scales, and energy, calculated using Kolmogorov's arguments as functions of time.

It is clear from figure 5.6(b) that the Kolmogorov length scale calculated using equations 5.17 and 5.16 is smaller than the resolution of the simulation. Under the ILES approach, not all the length scales are resolved. Because we are interested in the behaviour of inertial particles that are influenced by the smallest scales in the flow, unresolved flow scales are problematic. The unresolved scales would make large contributions to the physical behaviours of the particles,

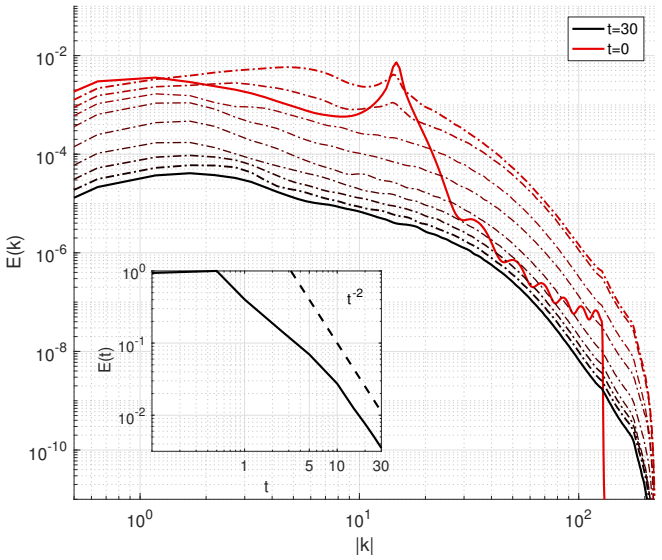


Figure 5.4: *The energy spectrum $E(||\mathbf{k}||,t)$ for a simulation with a velocity field initially given by $\mathbf{u}(x,0)$ in equation 5.15 is plotted for a series of instances in time. A spectrum is plotted for every 3 simulated seconds, from $t = 0$s (red solid line) to $t = 30$s (black solid line). Intermediary spectra are plotted with dashed lines that progressively approach the colour and position of the spectrum at $t = 30$s. The flow had periodic boundary conditions in all directions and gravity was not active, $\mathbf{g} = 0$.* ***inset*** *The total energy in the system decays as a power law beyond $t = 10$s, so that $E(t) \approx \lambda t^{-2}$, where λ is an arbitrary constant.*

from collision rates to clustering. The solution to the problem is to fix the Kolmogorov length scale throughout the simulation. MOBILE is capable of resolving scales down to approximately three times larger the grid resolution. Using this as our limiting factor, three times the grid resolution will be used to define the Kolmogorov length scale of the flow. The remaining micro-scales can be calculated relative to the fixed Kolmogorov length scale. Effectively the simulation has an artificial viscosity which is given by,

$$\begin{aligned} \nu(t)^3 &= \eta^4 \epsilon(t) \\ \nu(t) &= (\eta^4 \epsilon(t))^{\frac{1}{3}}, \end{aligned} \tag{5.18}$$

where equations 5.17 have simply been rearranged.

Although the effective viscosity resulting from scaling around the smallest resolved scales is unnaturally high for the specific contexts being investigated, it has little bearing on the behaviour of the particles. In our case it is unimportant whether the entire flow field accurately reflects that of a realistic cloud or river, as the particles are influenced by accelerations that act over a length scale relatively close to their own radius.

Given our definition of the Kolmogorov length scale, the particle behaviour is best categorised using the the resolution of the simulation. Therefore, the resolution of the simulation has a

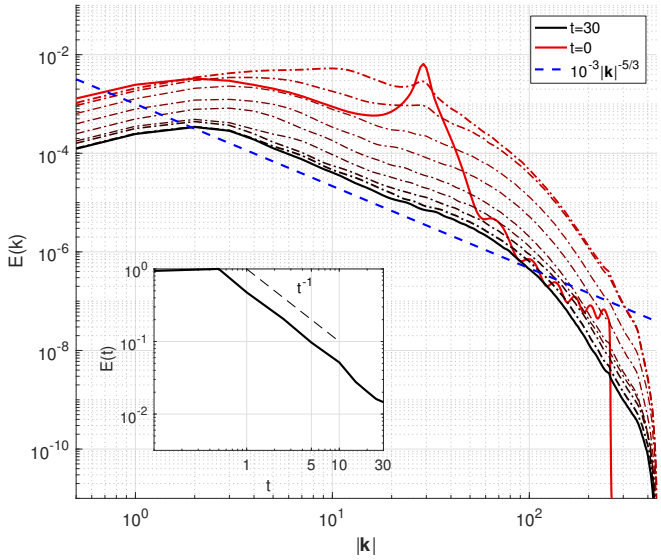


Figure 5.5: *Figure showing the same information as figure 5.4, but in this case the flow was bounded in the vertical with gravity equal to $g = -9.81$.* ***inset*** *The decay of the flow is slower when gravity is included so that $E(t) \approx \lambda t^{-1}$, where λ is an arbitrary constant.*

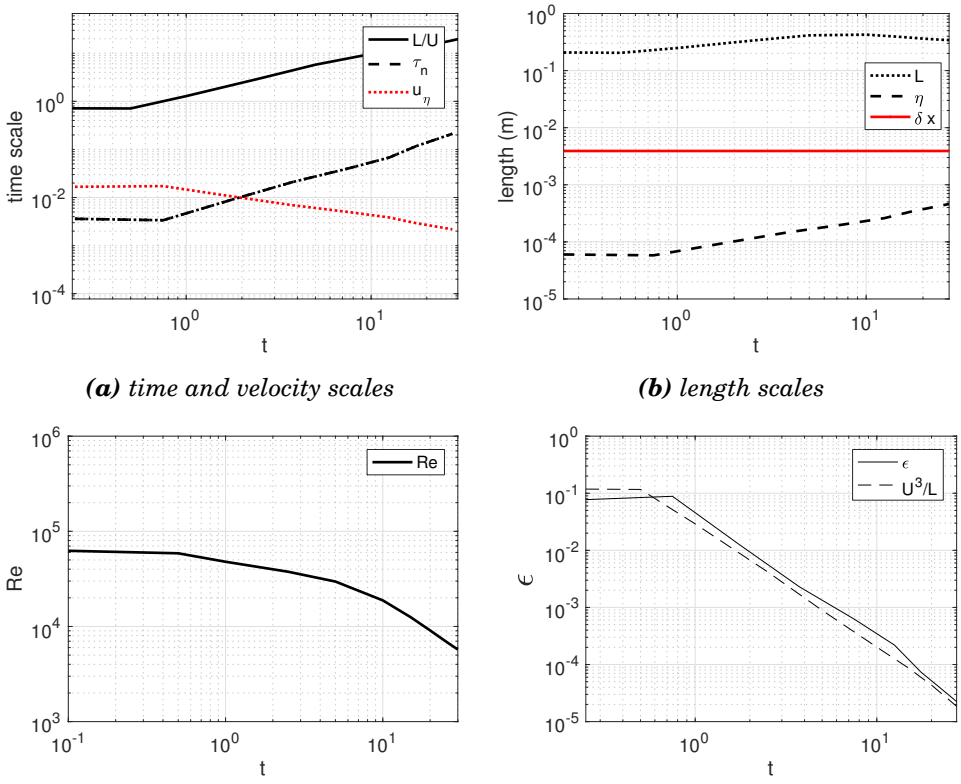
bearing on the particle Reynolds number, Re_p. We recall that the Maxey-Riley equation has a practical region of validity defined by $Re_p < 5$, although the strict mathematical limit, due to assumptions made in the derivation of the equation, is actually $Re_p \ll$. Our simulations are conducted in a $1m^3$ box, consisting of 256^3 cells, leading to a spatial resolution of $\Delta x = 1/256 \approx 0.0039$. Provided the particles investigated have diameters many times smaller than Δx, then the requirement on the particle Reynolds number, $\frac{2r_p\rho_f(v-u)}{\mu} < 5$, will be satisfied.

Given the fluid velocity field and a method for categorising particle behaviours within it, in the following section we discuss methods for categorising clustering behaviours.

5.3 Categorising clustering

Before we determine a method to quantify clustering, let us familiarise ourselves with what the term means. The clustering of the particles is the degree to which they group together in certain regions of the flow. Inertial particles that are more dense than the fluid cluster because they are ejected from vortical flow features, causing them to group together between them. Figure 5.6 shows a planar slice through the probability distribution function predicted by the transfer operator. The simulation was initialised with the velocity field given in equation 5.15, gravity was inactive and the boundary conditions were periodic. The density ratio R was akin to sand in water and equal to 0.4.

Some subtle clustering can be seen in all four images of figure 5.7. Some regions of the flow



(a) *time and velocity scales*

(b) *length scales*

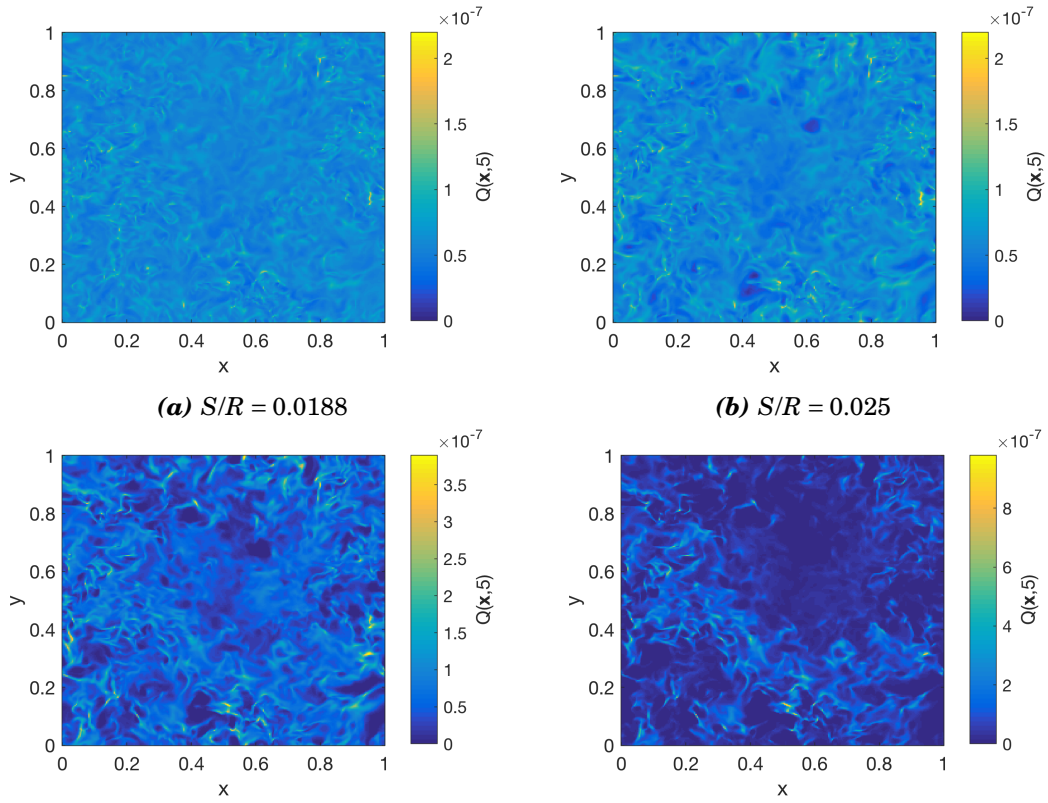
(c) *Reynolds number*

(d) *mean energy dissipation rate*

Figure 5.6: *The characteristic time and length scales, and the Reynolds number have been calculated for the bounded test flow using the energy spectra in figure 5.5. Each of the sub-plots show different characteristic flow measures. Similar results are obtained for the unbounded case. The Kolmogorov length scale can be seen to be considerably smaller than the smallest scales in the simulation in (b). In (c) we have plotted the energy dissipation rate ϵ and an approximation given by U^3/L [6].*

are more likely to contain particles than others, shown by the fact that the value of the PDF of particle positions is higher in certain regions.

The degree of clustering displayed is clearly dependent on the parameter S/R. As S/R increases, so does the prevalence of more densely populated regions. The scale over which the clustering occurs is also noticeably dependent on S/R. For S/R equal to 0.25 there is a clear large scale pattern to the clustering regions. Conversely, for $S/R = 0.0188$ the clustering is relatively homogenous throughout the domain. Given a picture of what is meant by clustering, let us define it quantitively. As in the third chapter, we would like to use the trajectories of individual particles to verify the transfer operator. The degree of preferential concentration and rate of sedimentation predicted by the transfer operator should match the particle simulations. Before a comparison can be made, we must define a clustering measure, but defining a clustering measure that can be used for both simulation types is difficult. The information from the two



(a) $S/R = 0.0188$ ***(b)*** $S/R = 0.025$

(c) $S/R = 0.125$ ***(d)*** $S/R = 0.25$

Figure 5.7: *The images show snapshots of the PDF of particle positions predicted by the transfer operator in the* $x-y$ *plane at* $z = 0.5$*. The transfer operator predicts increasing particle clustering as the particle parameter* S/R *is increased. The spatial scale over which the clustering occurs also increases so that, in (d), large areas of the flow are devoid of particles. Highly populated regions are represented by yellow streaks. Note that the colour scale is different in each image to reflect the full range of the PDFs. The flow was fully developed when the snapshot was taken after* 5 *simulated seconds.*

simulation types is very different. The transfer operator method produces a continuous, but discretely sampled, representation of particle positions in space. A simulation of many individual trajectories produces a noisy and statistically non convergent distribution of particles.

The seemingly obvious solution is to produce a cell wise representation of the particle distribution. First, we bin the particle positions over the same Cartesian grid used by the transfer operator. The result is an array containing the number of particles per transfer operator cell, which we denote with $M_p(x)$, where X is a discrete variable representing the cell index. The PDF of particle positions predicted by the transfer operator is given by $Q(X)$. To allow comparison

between $Q(X)$ and $M_p(X)$ we normalise both as follows,

$$\widehat{Q}(X) = \frac{NQ(X)}{\sum_X Q(X)}, \qquad \widehat{M}_p(X) = \frac{NM_p(X)}{\sum_X M_p(X)}, \tag{5.19}$$

where N is the number of cells. A circumflex is used to indicate that the variable has been normalised. The normalisation ensures that the mean of the distributions is equal to 1.

In the third chapter we binned the particles from the particle simulation over a spatial grid. We were then able to successfully compare the results produced by the transfer operator with simulations of large numbers of particles. However, the space discretised in the third chapter was two dimensional. Many more particles are required in a 3-dimensional space to reach statistical convergence than in 2 dimensions. Therefore, in 3-dimensions we suffer more from the presence of noise in the particle distribution. To reduce the influence of the noise in the particle distribution we use the centre weighted smoothing used in chapter 3. In 3-dimensions the smoothing is conducted using,

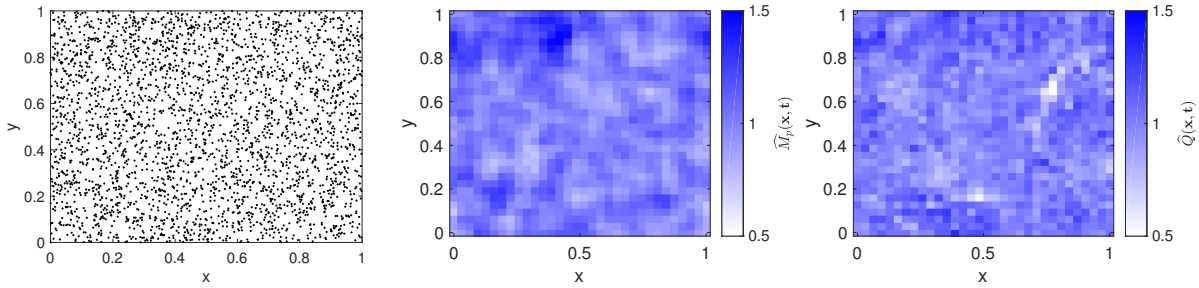
$$M_p^{(i,j,k)} = \frac{2M_p^{(i,j,k)} + M_p^{(i-1,j,k)} + M_p^{(i,j-1,k)} + M_p^{(i+1,j,k)} + M_p^{(i,j+1,k)} + M_p^{(i,j,k-1)} + M_p^{(i,j,k+1)}}{8} \tag{5.20}$$

where $M_p^{(i,j,k)}$ denotes the number of particles in cell (i,j,k) of the discretisation.

As a test case let us look at the results of a simulation with $N = 256^3$ cells. We use equation 5.15 as an initial condition for the flow field and simulate $N_p = 10^5$ particles in a triply periodic domain. We bin the particles at two different resolutions: at the resolution of the simulation where there are 256^3 cells, and secondly over a grid of 64^3 cells.

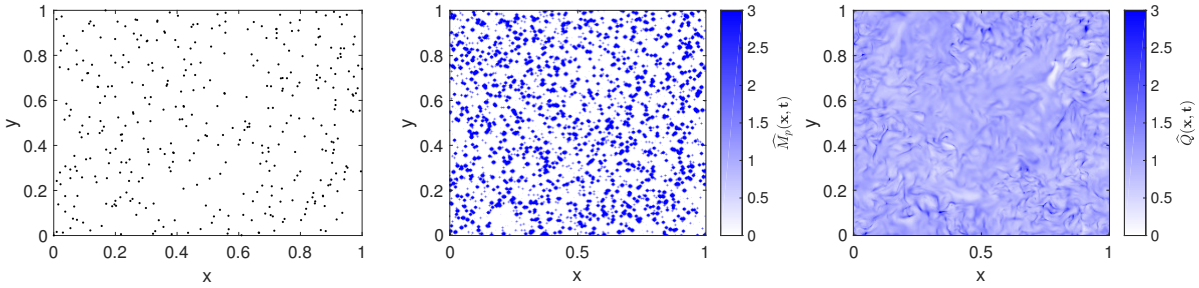
Figure 5.8 shows the particle positions before and after binning in a planar slice through the middle of the domain. The slice is orientated in the $x-y$ plane and is one cell thick in the vertical (i.e. z ranges between $31\Delta z \le z < 32\Delta z$). The snapshot was taken after 5 seconds. Figure 5.8(a) shows the actual particle positions in the plane before binning and figure 5.8(b) shows the number of particles per cell after binning. After binning, one iteration of spatial smoothing using equation 5.20 is conducted. Figure 5.8(c) shows the PDF predicted by the transfer operator. Whilst some features appear in both figure 5.8(b) and figure 5.8(c) the agreement is poor. The features of the flow field that create the majority of the clustering (the smallest vortices) are close in scale to the grid. Binning over a coarse grid misses the effect of the smallest flow features and therefore much of the clustering.

Figure 5.9 shows the other extreme. When binning over a high resolution grid, there are too few particles in the simulation for statistics to converge sufficiently. The clear clustering behaviour predicted by the transfer operator is completely missed by the particle simulation.



(a) *particle positions* ***(b)*** *after binning* ***(c)*** *transfer operator*

Figure 5.8: *In order to compare the transfer operator predictions to the positions of Lagrangian particles, we must bin the Lagrangian particle positions in space. A snapshot of particle positions in a planar slice in the $x-y$ plane at $z \approx 0.5$, $t = 5$, and $S/R = 0.0250$ is shown for both the Lagrangian particle simulation and the transfer operator simulations. The choice of resolution for the binning of particles is difficult: too coarse a grid and the clustering is missed, too fine a grid and the statistics are extremely noisy. In this figure the particles were binned over a grid of $N = 64^3$ cells.*



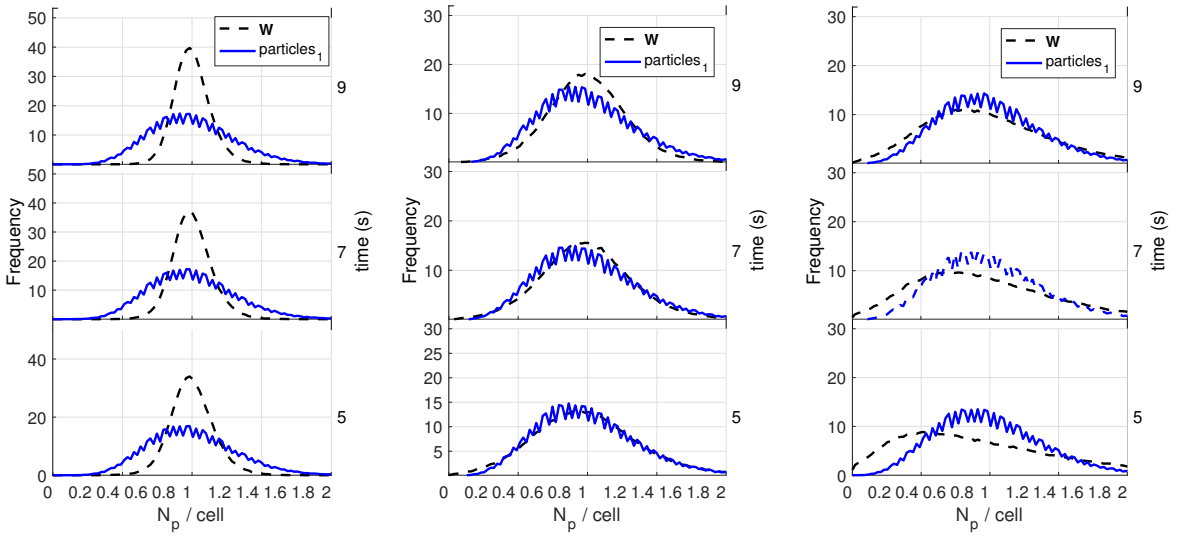
(a) *particle positions* ***(b)*** *after binning* ***(c)*** *transfer operator*

Figure 5.9: *The same information is shown here as in figure 5.8. However, the grid over which the particles are binned is much finer with $N = 256^3$. With such a fine grid, too few particles were simulated to reach statistical convergence.*

It is clear that an alternative approach to categorise clustering is required. Let us try to compare global statistics on the clustering. As in the second chapter, we use histograms of the PDFs as our comparator. The space of the number of particles per cell is discretised. A count is then assigned to each discrete bin to produce a histogram, or distribution. Processing the data in this way is useful because the characteristics of the histograms can be used to characterise clustering. For example, the more the histogram is skewed to the left the greater the degree of clustering. Figure 5.10 shows the histograms for three values of S/R: $S/R = 0.0188$, $S/R = 0.1250$ and $S/R = 0.2500$. Distributions are produced for three instances in time to show how the clustering behaviour changes with the decay of the turbulence.

There is obvious disparity between the distributions produced from the transfer operator simulation, and those produced from the particle simulation. First let us focus on the transfer

operator distributions. At $S/R = 0.0188$ there is very little clustering. The distributions have modes close to one and show relatively little skewness. At $S/R = 0.125$ the degree of clustering predicted by the transfer operator is larger. Wide distribution tails indicate that there are some bins with many more particles in than others. For $S/R = 0.25$, even more clustering can be is predicted by the transfer operator. All distributions are strongly skewed to the left and the tails have spread further still.



(a) *S/R=0.0188* ***(b)*** *S/R=0.1250* ***(c)*** *S/R=0.2500*

Figure 5.10: *In order to compare the clustering predicted by the transfer operator with the clustering displayed by particles in the Lagrangian particle simulations, we plot frequency distributions over the number of particles per cell. The further the distributions are skewed to the left, the greater the degree of clustering. Discrepancies between the frequency distributions predicted by the transfer operator (dashed line) and the individual particle trajectories (solid blue line) are due to insufficient numbers of particles in the simulation. As a result, the Lagrangian particle distributions do not change for the particle simulations. An alternative approach to characterise clustering in the particle simulations will be used.*

Now we turn to the distributions produced from the particle simulations which tell a different story. Whilst there is a small increase in clustering as the value of S/R increases, it is extremely subtle. Randomness in the particle distribution dominates and much of the clustering is missed.

We can successfully categorise the clustering predicted by the transfer operator using the distributions, but are unable to do the same for the particle simulations. To capture the clustering shown in the particle simulations we need a further approach. Following [7], we introduce the correlation dimension D. Let $P(Y)$ denote the probability of finding two particles closer together than Y. For $i = 1,2,3...N_p$, where N_p is the number of particles,

$$P(Y) = \frac{2\sum_i \sum_j \Theta(Y - ||x_i - x_j||)}{N_p^2 - N_p}, \quad j > i, \tag{5.21}$$

where x_i and x_j are the positions of two particles, Θ is the Heaviside function and N_p is the number of particles in the simulation. For small Y, $P(Y) \propto Y^D$ and D can be found by taking logarithms,

$$D \approx \frac{\log_{10}(P(Y)) - \log 10(\lambda)}{\log_{10}(Y)}, \tag{5.22}$$

where λ is some proportionality constant. To find D in practice, we fit a power law to $P(Y)$ for $0 < Y/\eta < 5$. In [30] the power law was fitted for $0.1 < Y/\eta < 10$, but we find including larger values of Y dramatically changes the value of D. For $Y > 5$ the distribution no longer follows a power law. We will show in the following section that the correlation dimension is extremely successful for quantifying the clustering of the particles.

5.4 Results

We have conducted simulations in both the unbounded flow and the flow bounded in the vertical using parameters found in table 5.1 and table 5.2 in section 5.2.4. For each parameter set, a simulation was conducted using the active tracer equation introduced in second chapter, the full Maxey-Riley equation solved in the second, and using the transfer operator introduced in the third. The results are split into three sections. First, a comparison is made between the transfer operator and the active tracer simulations in the unbounded flow. Then a similar comparison is made in the bounded flows, where the sedimentation rates of the particles provides us with further comparators. In the third section we explore the influence of the history force on the degree of clustering and the sedimentation rates of the particles.

5.4.1 Unbounded flow

Our first observation concerns the dependence of the particle clustering on the energy in the flow. Figure 5.11 shows the distribution over the number of particles per cell predicted by the transfer operator. The value of S/R was equal to 0.25. Distributions are plotted for a series of instances in time to show how the clustering decays along with the flow. Initially, when the energy in the flow is maximal, the clustering is extremely pronounced. After 10 seconds, the particles have returned to a more random distribution.

A random distribution categorises the behaviour of passive tracers that follow fluid streamlines. Passive tracers in a turbulent flow produce a distribution close to Poissonian because the turbulent vortices mix them randomly. As the turbulent energy in our flow decreases, the Stokes number of the particles decreases, and the particles behave more like passive tracers. In fact, the turbulent eddies serve to homogenise the particle distribution as the simulation progresses.

The question is by how much does the clustering decay relative to the energy in the flow. Let us turn our attention to the simulations of many individual particle trajectories. The degree of clustering in the particle simulations can be measured using the correlation dimension, D,

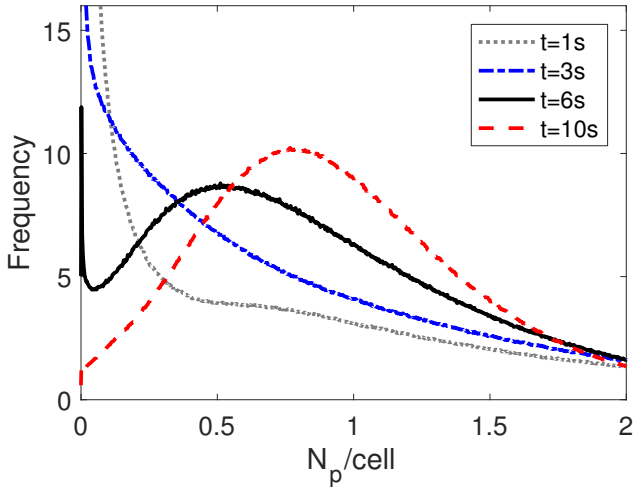


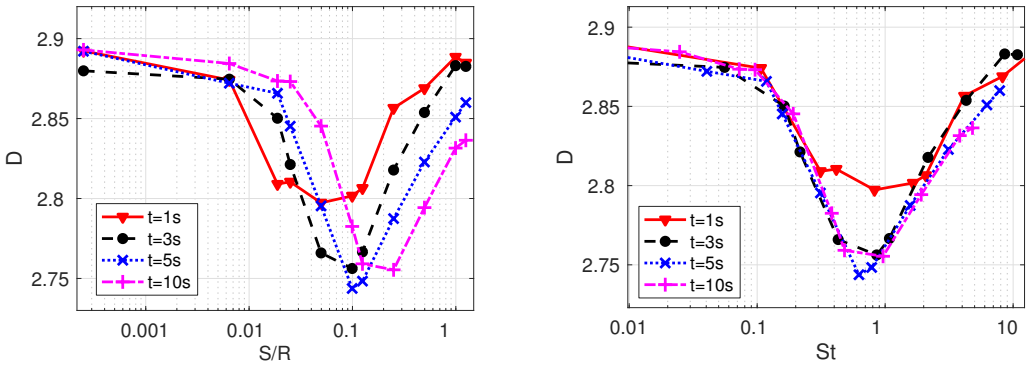
Figure 5.11: *Frequency distributions, over the number of particles per cell, are plotted to investigate particle clustering predicted by transfer operator. Distributions are shown for a range of times for a fixed value of $S/R = 0.25$. The left skewness of the distributions indicates the degree of clustering. Over time, the energy in the flow decays and the degree of clustering decays with it.*

defined in equation 5.22. Figure 5.18 shows the correlation dimension calculated for the particle simulations for four instances in time: $t = 1$, $t = 3$, $t = 5$, and $t = 10$ seconds. The correlation dimension D is plotted as a function of S/R in (a), and as a function of $St = \frac{SU(t)}{R\eta}$ in (b). Because we want a clear definition of a timescale for our flow that is not affected by complications of the effective viscosity discussed earlier, we have chosen to normalise the particle parameters based on the Kolmogorov length scale and the root mean square velocity $U(t)$. We know that $\tau_n \propto \frac{\eta}{U(t)}$, and we find defining a Stokes number based on $\frac{\eta}{U(t)}$ produces good results. By plotting the correlation dimension as a function of St, rather than as a function of S/R, the curves collapse onto one another.

The way the curves collapse so readily onto one another indicates the importance of the parameter $U(t)$ as a method for categorising particle behaviour. It appears that the degree of clustering of active tracer particles is linearly dependent on $U(t)$. By extension, it appears that the degree of clustering is proportional to $\sqrt{E(t)}$.

In figure 5.18 there is an obvious value of St that produces maximal clustering. Maximum clustering (a minimum value of D) occurs for approximately $St = 1$ in all cases. In fact, it is well known that particle clustering in turbulent flows reaches a peak close to $St = 1$ (i.e where the particle response time is close in value to the Kolmogorov timescale) [7, 101].

The mono-modal clustering with St is observed for the particle simulations, but not for the transfer operator. Figure 5.13 shows the distribution of number of particles per cell for a series of representative Stokes numbers. The degree of clustering increases continuously for increased Stokes number, displaying no peak at $St = 1$. Such discrepancy between the particle simulations



(a) *The correlation dimension, D, is plotted as a function of S/R.*

(b) *The correlation dimension, D, is plotted as a function of St.*

Figure 5.12: *The correlation dimension, D, is used to measure clustering in the Lagrangian particle simulations. The correlation dimension is plotted both as a function of S/R and St to show that clustering is dependent on St and not just S/R. When the root mean square velocity, U, is taken into account in the Stokes number, the correlation dimension plots collapse onto the same curve. The most extreme clustering occurs for $St \approx 1$ for all time.*

and the transfer operator predictions indicates a region of validity for the transfer operator similar to one found in the third chapter. However, to draw such conclusions we desire a method of more direct comparison between the methods.

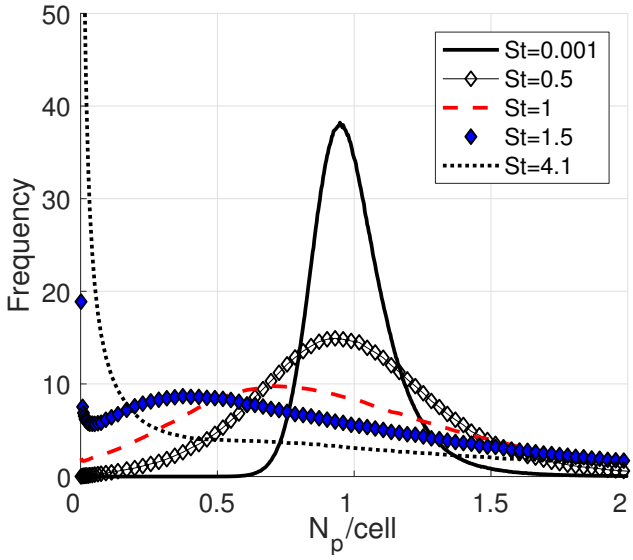
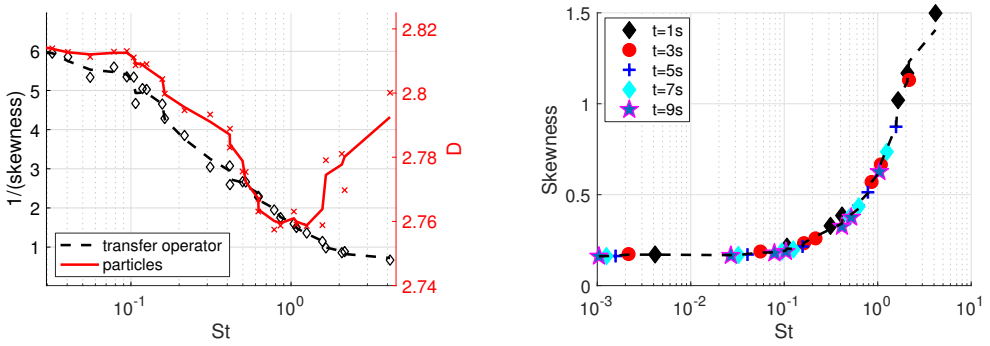


Figure 5.13: *Frequency distributions over the number of particles per cell predicted by the transfer operator, are shown here for a range of Stokes numbers. The higher the Stokes number, the greater the skewness of the distributions, and greater skewness of the distributions corresponds to a greater degree of clustering.*

To allow comparison between the transfer operator and the particle simulations, we measure the skewness of the distributions in figure 5.13. The skewness of each distribution is plotted as a function of St in Figure 5.14(b). We recall that increased skewness of the distribution indicates more clustering. In the third chapter we demonstrated that the efficacy of the transfer operator diminishes around $S/R = 0.1 \approx St = 1$. The limit of validity is therefore no surprise, but reiterates that the region of usefulness of the method is limited to small Stokes numbers.

In figure 5.14(b) we compare the clustering predicted by the transfer operator directly with the clustering found in the large particle simulation. To do so, we plot $1/Sk$, as a function of Stokes number on the left vertical axis, where Sk is the skewness of the distributions predicted by the transfer operator. On the right vertical axis, we plot the correlation dimension as a function of St. Both axes have been scaled so that the minimum clustering displayed by both is coincident. A moving average has also been plotted for both data sets to guide the eye. The noise in the data arises from the data being taken from a wide range of times within the simulations.



(a) *Comparison between particle simulation and transfer operator.* ***(b)*** *Skewness of transfer operator distributions.*

Figure 5.14: *(a) Clustering is measured differently for each modelling approach. Skewness of distributions over the number of particles per cell is used for the transfer operator simulations, and the correlation dimension is used for the Lagrangian particle simulations. The clustering predicted by the transfer operator (black diamonds) is plotted together with the clustering predicted by the particle simulations (red crosses). Both modelling approaches display the same trend up to $St \approx 1$, beyond which the transfer operator continues to predict increases in particle clustering for increasing St. (b) The skewness of the frequency distributions predicted by the transfer operator increase with St, but do not display the expected mono modal behaviour as in figure 5.18.*

There is a region in figure 5.14 where the rate of increase of clustering is similar for both simulations. Beyond the expected peak of the clustering, $St = 1$, the transfer operator begins to over predict the clustering.

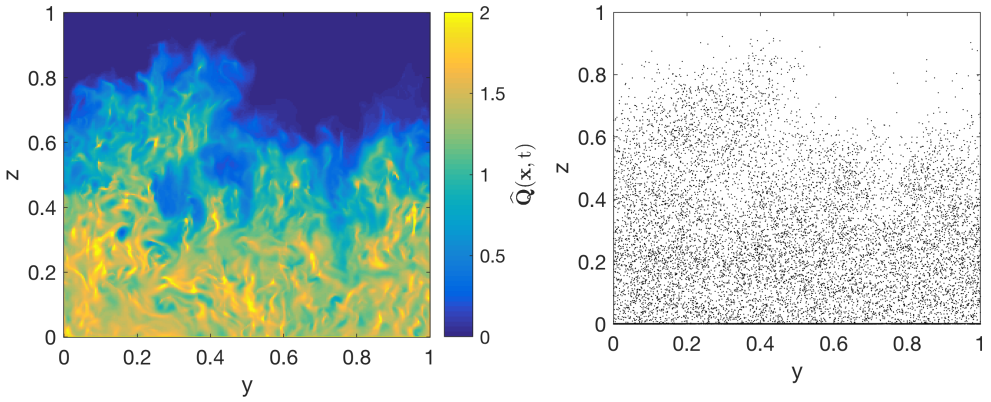
The over prediction of clustering for large Stokes numbers is due to the more massive particles ignoring the smallest features in the flow field. Rather than accelerate with the flow, they deviate less from their original trajectories. Similar behaviour can be seen in figure 2.6 in the third chapter.

In the third chapter we investigated a simple two dimensional flow field with two periodic vortices. In that case, instead of particles being attracted to the separatrices, the clustering occurs along a wider band around the separatrix. The result is that the attractive regions between the vortices are less sharp for more massive particles and clustering is reduced. It is surmised that a similar behaviour is happening in the more complicated flow field used in this chapter. The behaviour arises from particles carrying large amounts of momentum from vortex to vortex. Each new vortex is unable to impose itself on the particles enough to considerably change the particle momentum. The transfer operator approach on the other hand, takes into account only the current position of the particle to predict its future state. As such, a particle does not arrive in a box with a memory of any previous vortices and the transfer operator is unable to replicate the peak clustering behaviour. In the case of the transfer operator, larger Stokes number simply imply increasingly large vortex ejection rates, and consequently increasing clustering. That is not to say that the transfer operator is unsuccessful, rather that it has a region of validity.

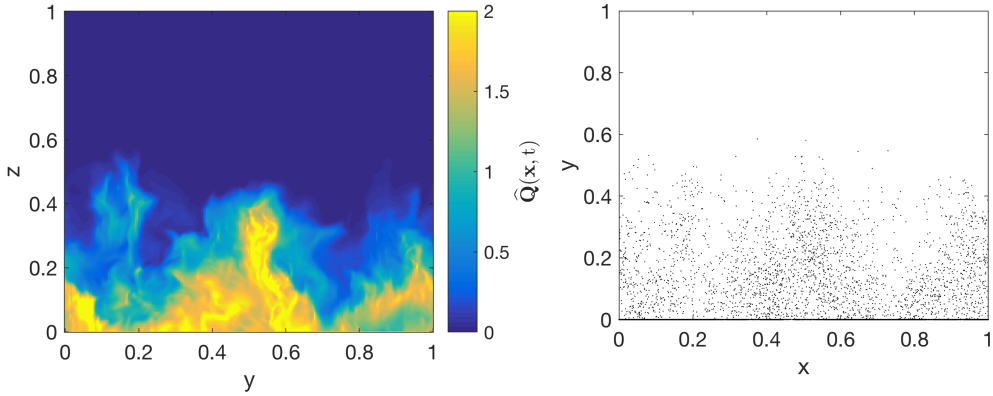
5.4.2 Bounded flow: sedimentation

In the second test case, the influence of gravity is investigated. Preferential concentration of particles is known to affect the sedimentation rate of particles. In regions where there are many particles, the sedimentation rates are reduced [102]. The ability to replicate the sedimentation rate of inertial particles in the continuum limit, allows the influence of preferential concentration on sedimentation rates to be investigated. Here we show that the transfer operator exactly predicts the sedimentation rate of the particles in the turbulent test flow.

First let us glean a general understanding of the sedimentation behaviour of the particles. Figure 5.15 shows two snapshots of a simulation with $S/R = 0.0250$ at $t = 5s$ (figure 5.15(a)) and $t = 10s$ (figure 5.15(b)). The left hand panel in both cases shows the prediction made by the transfer operator, whilst the right hand panel shows the distribution of particles from the particle simulation. In both cases the sedimentation rate is well matched. Further, the particle distributions match in the horizontal direction too, demonstrating that the variable sedimentation produced by variability in the turbulence is captured by both simulations. For a quantitative assessment of the sedimentation rates, we average the particle concentrations over vertical slices of thickness Δx. The result is the average concentration of particles as a function of depth. The process is repeated every second during the simulations for both the transfer operator and the particles to generate a matrix containing average particle concentrations for a number of depths and times. The matrix can be plotted as an image, that shows as a colour gradient, how the spatial distribution of the particles changes throughout the simulation. We recall the variables $\widehat{M}_p(\mathbf{x},t)$ and $\widehat{Q}(\mathbf{x},t)$ from earlier sections. The variable $\widehat{M}_p(\mathbf{x},t)$ describes the number of particles per cell from the particle simulation, and $\widehat{Q}(\mathbf{x},t)$ describes PDF of particle positions predicted by the transfer operator. Averaging in horizontal slices produces two variables that are simply functions of depth and time, $\widehat{M}_p(z,t)$ and $\widehat{Q}(z,t)$.



(a) Left: $\widehat{\mathbf{Q}}(x = 0.5, t = 5)$ *and the* ***Right:*** *distribution of particles for* $0.45 < x < 0.55$ *after 5 seconds.*



(b) Left: $\widehat{\mathbf{Q}}(x = 0.5, t = 10)$ *and the* ***Right:*** *distribution of particles for* $0.45 < x < 0.55$ *after* 10*s.*

Figure 5.15: *The transfer operator prediction of sedimenting particles (left) compares well with the actual locations of particles from Lagrangian particle simulations (right). The snapshots were taken after* 5*s (top) and* 10*s (bottom).*

Figure 5.16 shows the distribution $\widehat{M}_p(z,t)$ compared with $\widehat{Q}(z,t)$ by way of contours. Three representative contour values are chosen, $\widehat{M}_p(z,t) = [0.001, 0.1, 0.5]$. The centre of mass is also plotted as a function of time for both the transfer operator and the particle simulations. Initially the centre of mass is exactly halfway up the tank as the particles are homogeneously distributed in space. By the end of the simulation the particles have sedimented out and the centre of mass is at the bottom of the tank. For each of the contours and the centre of mass, the transfer operator and the particle simulations produce excellent agreement.

It is clear that the transfer operator does a good job of representing the sedimentation statistics of the whole ensemble. What is less clear is whether the clustering of the particles is accurately captured. An advantage of the sedimentation simulations is that the particle concentration quickly increases at the bottom of the tank. The increase in concentration greatly improves the statistical quality of the particle distribution for a certain region of the flow. Better

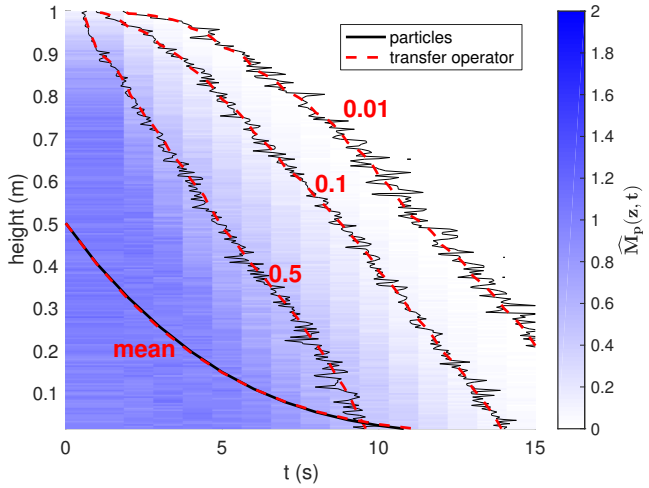


Figure 5.16: *The distribution of the particles in the vertical predicted by the transfer operator match exactly with the direct simulation of a large number of Lagrangian particles. The numbers (0.5, 0.1, 0.01) represent the level sets of the scaled particle density field. For all level sets, the agreement between the particle simulation and the transfer operator simulation is excellent.*

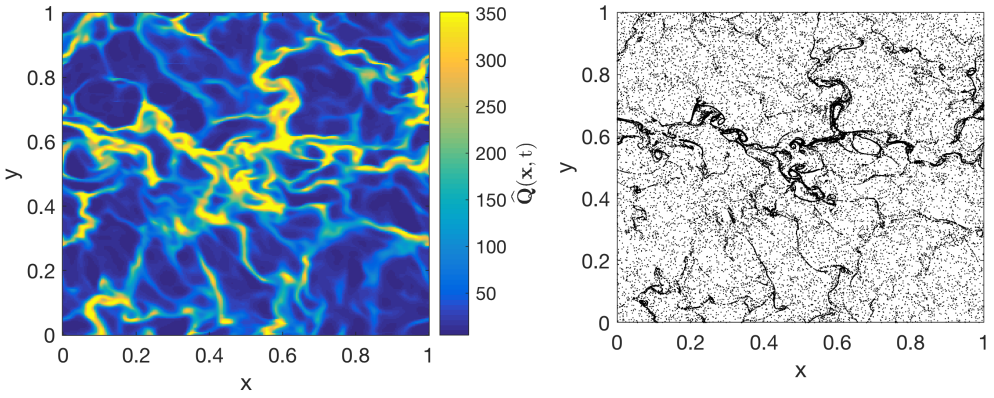
comparisons between the transfer operator and the particle simulations can be achieved in the regions near the bottom of the tank.

Figure 5.17 shows two snapshots of the bottom few centimetres of the tank where the particle concentration is high. Again $St/R = 0.0250$ and the snapshots are taken at $t = 5s$ (figure 5.17(a)) and $t = 10s$ (figure 5.17(b)). There is very strong clustering behaviour at the bottom of the tank where the particles are trapped vertically. The patterns created by high and low particle concentrations are well matched in both cases with features appearing in identical locations. Of course, if both modelling approaches are working as they should, then the two figures should match: the fact that they match so well, gives us confidence that the transfer operator approach is successfully capturing the inertial properties of the particles in the turbulent flow.

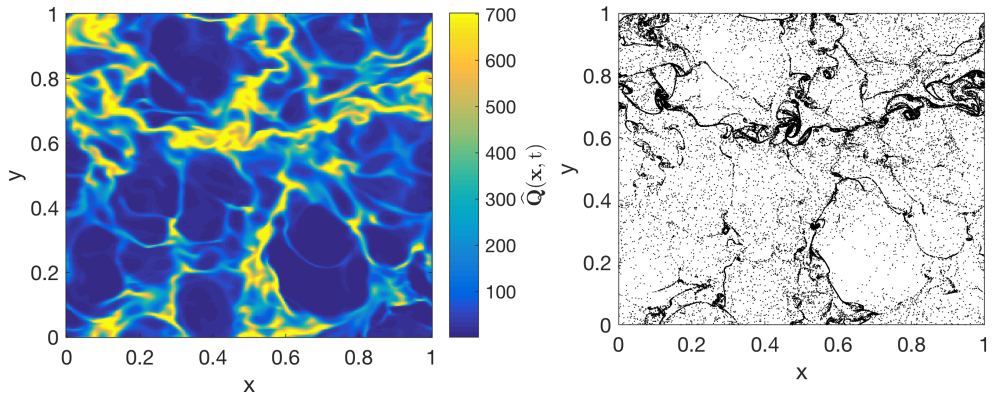
An interesting observation of the patterns produced in figure 5.17 are the large scale nature of the features around which the particles cluster. There are features of the flow which are acting over a much smaller length scale than the shapes produced by the particles. Such clustering behaviour implies that the particles are successively ejected from smaller vortices to become aligned in the separatrices between the larger more persistent ones.

5.4.3 Including history

In this section we investigate the influence of the history force on particle clustering and sedimentation rates in turbulent flows. Whilst there are a number of studies investigating the relevance of various particle forces, there are few studies that investigate the influence of the Basset force on particle clustering. Recently, however, two studies appeared, one from Olivieri [31] and one



(a) Left: $\mathbf{Q}(z = \Delta x, t = 5)$ *and the* ***Right:*** *distribution of particles for* $z < 0.1$ *after* 5*s*.



(b) Left: $\mathbf{Q}(z = \Delta x, t = 10)$ *and the* ***Right:*** *distribution of particles for* $z < 0.1$ *after* 10*s*.

Figure 5.17: *The bottom few centimetre of the tank are densely populated with particles and provide an opportunity for visual comparison of clustering features. The distribution of particles at the bottom of the tank predicted using the transfer operator (left) compares well with the actual particle positions from direct simulation of particle trajectories (right). Features in the clustering regions of the particles match qualitatively between the two methods.*

from Daitche [30] which investigated clustering in the presence of history. Both found the history force reduced clustering, but disagreed about its significance for different parameter regimes. Daitche found that the history force greatly reduced clustering for small density ratios but was insignificant for larger ones such as in raindrop formation. Olivieri on the other hand found that clustering was reduced for all density ratios provided St was large enough. We aim to ascertain whether the influence of history is indeed reduced for large density ratios.

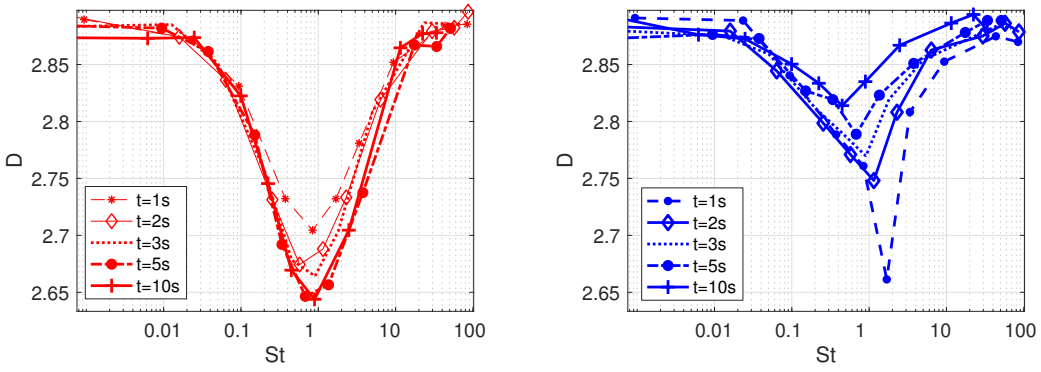
We are also motivated to define the region of validity of the transfer operator. Including the history force in the transfer operator method is extremely complicated. It is important to identify when history is important so that we can define the region of the parameter space for which the transfer operator can be used. Where history is significant other modelling approaches should be adopted.

We also investigate the role of the history force in a system of sedimenting particles. We wish to understand its influence on the sedimentation rate of particles, and how its influence changes as a function of the energy in the flow. In the simple case of a spherical particle falling under gravity in a quiescent fluid, the influence of the history force is significant in reducing the terminal velocity of the particle [103]. It is expected therefore, that history will also reduce particle sedimentation rates in turbulent flows.

The simulations in the previous sections have been repeated with the history force included using the parameter values in table 5.1 and table 5.2. History is included using the numerical method detailed in the second chapter. When coupled with the requirements of the fluid solver, the memory requirements of including history are very large. To make simulations tractable, we use a shorter history window than in the first chapter. The time for over which the history integral is evaluated, t_{win}, is set to 1 second rather than the 2 seconds used in the second chapter. Beyond 1 second, the tail of the history force is approximated using Hinsberg's exponential kernel [63]. When history is included, the step size of the method is fixed and the other discrete time parameters are defined relative to this: $\Delta t_p = \Delta t_f = 2\Delta t$.

We first compare our results to the work of Daitche; for this we recall the correlation dimension D. The correlation dimension provides a single value that characterises the degree of clustering displayed by the particles. Daitche found that the clustering peaked (D was at a minimum) at approximately $St = 1$, both with and without history. Further, for small density ratios Daitche found that particle clustering was significantly reduced when history was included, but had a lesser effect as ρ_p/ρ_f increased.

Figure 5.18 shows the correlation dimension, D, plotted as a function of St for $R = 10^{-3}$. A density parameter of $R = 10^{-3}$ is representative of water in air. Figure 5.19 shows the correlation dimension D as a function of St for $R = 0.4$, which is representative of sand in water.



***(a)** without history* ***(b)** with history*

***Figure 5.18:** The history force is known to influence particle clustering. Here, the correlation dimension is plotted as a function of St with $R = 10^{-3}$, which is representative of water in air. History significantly reduces clustering as the simulation progresses.*

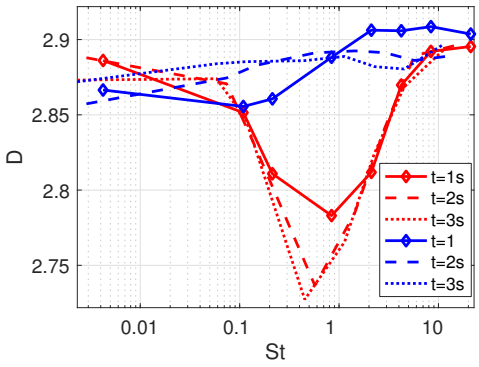


Figure 5.19: *When the particle and fluid densities correspond to sand in water, $R = 0.4$, clustering is hugely affected by the history force. The red lines show simulations conducted without history, and the blue lines show simulations where history was included. History destroys clustering for all St.*

Our results differ from those of Olivieri [31] and Daitche [30]. Daitche in particular found that particle clustering was reduced for smaller density ratios, but not for the higher density ratios of raindrop formation. In our simulations, significant clustering reduction can be seen for both the large density ratio case and the smaller density ratio case in our simulations. For the case of sand in water the clustering is destroyed by the presence of history.

A possible reason for the discrepancy is in the nature of the turbulence studied. We have investigated the clustering in a decaying turbulent flow as opposed to a forced turbulent flow. It is possible that the decaying nature of the flow has a bearing on the degree to which the history force influences particles. Let us remind ourselves of the form of the history force, F_B. The history force is an integral over all historical acceleration differences between the particle and the fluid, given by,

$$F_B = \frac{3R}{\sqrt{2\pi S}} \int_{t_0}^{t} \frac{1}{\sqrt{t-\tau}} \frac{d}{d\tau}(\mathbf{v}(\tau) - \mathbf{u}(\mathbf{x},\tau))d\tau. \tag{5.23}$$

As the fluid velocity decays, so does the size of the forces acting on the particles. Consequently, fluid accelerations from the particles history, when fluid velocities were higher, would be relatively more significant than acceleration differences from the present. In our flow, the characteristic velocity, U decays at a rate between t^{-1} and t^{-2} (see figures 5.4 and 5.5). The decay rate of U is greater than the decay rate of the history kernel which decays as $t^{-1/2}$. If accelerations from the particle's history are relatively larger than they would be in a non-decaying flow, then the behaviour of the particles could be significantly altered. In a forced turbulent flow, as in the one used by Daitche, U is constant throughout the simulation, meaning historical accelerations would not be relatively larger than accelerations from the particle's present.

Of course we are assuming here that the acceleration difference between the particle and the fluid decays with U which, although likely, is not necessarily the case. To investigate our

assumption, we have recorded the acceleration differences between the particles and the fluid for a simulation of 100 particles. The simulation consisted of $128 \times 128 \times 128$ fluid cells and the initial conditions of the velocity field were given by equation 5.15. The remaining parameters are the same as in table 5.1 and the boundary conditions are periodic in all directions. Figure 5.20 shows how the acceleration difference of the particles decays relative to U. Clearly the type of flow has

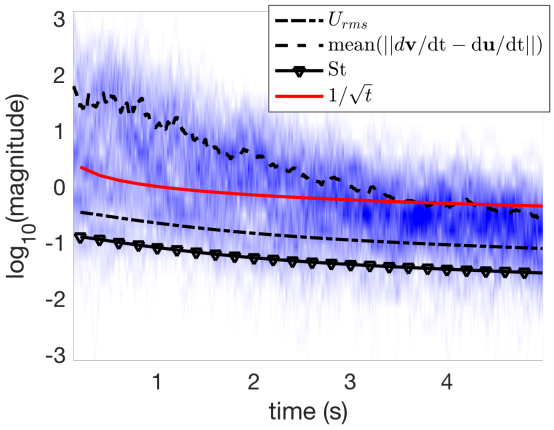
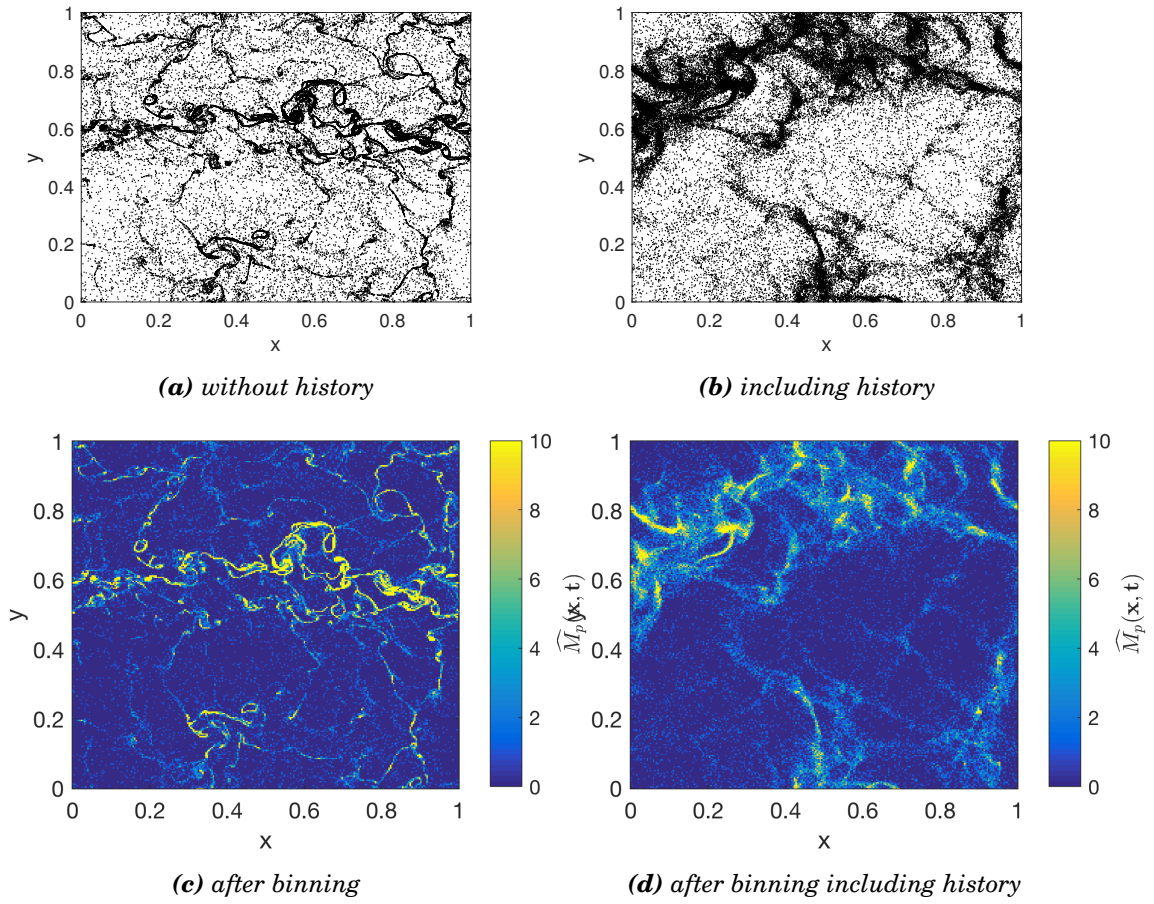


Figure 5.20: *History could be particularly potent in a decaying flow because historical accelerations are larger than accelerations in the present. The decay of the acceleration difference between the particles and the fluid is plotted here as a function of time for* 100 *particles. The average over all the particles is shown by the solid black line. Also plotted is the decaying U, the history kernel, and the Stokes number, to show how each decays relative to one another.*

a strong bearing on how influential the history force is likely to be. In decaying turbulence it should be more significant than in forced turbulence which is exactly what we have found.

Our second investigation is into the influence of history on sedimenting particles in the bounded flow. Here the density ratio used is representative of sand in water. As in the previous section, the regions at the bottom of the tank are densely populated with particles. Densely populated regions provide a good opportunity for visual comparison between the clustering when history is and isn't included. As we have found for the unbounded flow, clustering is significantly affected by the inclusion of the history force in the bounded flow. The reduction in clustering caused by history is clearly evident in figure 5.21. Figure 5.21 shows a comparison between the particle positions in the bottom centimetres of the tank when history is and isn't included. The snapshot was taken after 5 seconds, and the Stokes number of the particles was $St = 0.2$. The densely clustered regions are markedly different with and without history. With history, the densely populated regions are less sharp, indicating that the clustering is weaker. History also has an influence on the sedimentation rates of the particles. figure 5.22 shows the average vertical position of particles as a function of time for the three modelling approaches: active tracer particles, particles including history, and the transfer operator. History significantly reduces the sedimentation rate for $S/R = 0.25$ and $S/R = 0.125$, but is less influential for smaller S/R. In



(a) *without history* ***(b)*** *including history*

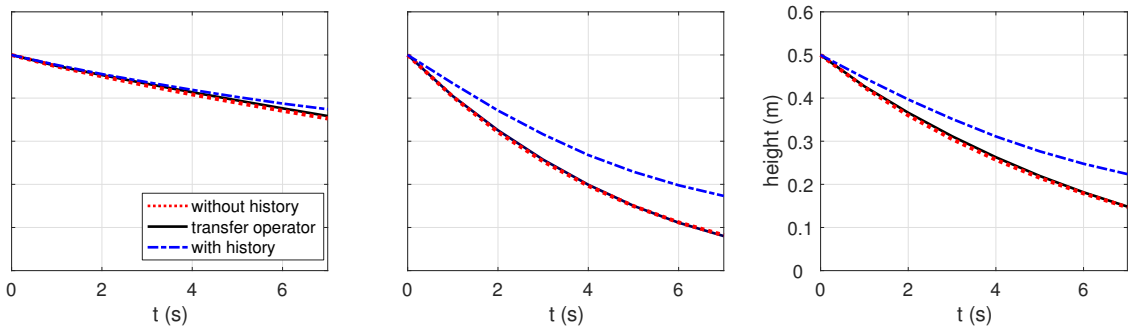
(c) *after binning* ***(d)*** *after binning including history*

Figure 5.21: *Again, the bottom of the tank in the sedimentation simulations provides a good opportunity for visual comparison between modelling approaches. Particle positions for* $z < 0.1$ *after* 5*s, both without history (left) and with history (right) are plotted. The particle parameters were given by* $S/R = 0.025$ *with* $R = 0.4$ *representative of sand in water. The Stokes number of the particles was* $St = 0.2$*.*

all cases the transfer operator accurately replicates the sedimentation rate of the active tracer particles. What is clearly evident is that the history force has an impact on particle behaviour for both density ratios investigated. The influence is significant for all St numbers for the small density ratio case of sand in air, but only significant for the raindrop formation case for $St > 0.1$.

5.5 Conclusions

The clustering and sedimentation behaviour of inertial particles has been investigated in a turbulent cloud-like flow. We have demonstrated that the transfer operator is effective in capturing the dynamics of the particles for certain regions of the parameter space defined by S and R. The measure of validity is given by the Stokes number St. For $St \leq 1$ the transfer operator approach appears to work well for active tracer particles. For larger particles, the inertial equation no



(a) *S/R=0.0188* ***(b)*** *S/R=0.1250* ***(c)*** *S/R=0.2500*

Figure 5.22: *The history force reduces particle sedimentation rates and its influence increases as a function of S/R. The vertical centroid of the particle positions is plotted as a function of time for three values of S/R. To demonstrate how effectively the transfer operator captures the dynamics of the active tracer, the sedimentation rates predicted by the transfer operator are also plotted on each plot.*

longer successfully describes the motion of the particles. Rather than display peak clustering for a value of $St = 1$, the transfer operator predicts that the degree of clustering will continue to increase with Stokes number.

In applications where the Stokes number is less than 1 and the history force is insignificant, our approach is valuable. Where particles are small and numerous, for example in the early stages of raindrop formation, an Eulerian particle description can provide significant computational savings. In such applications the deviation of the particles from the fluid flow are sufficiently small to allow the inertial equation to capture the important dynamics. Although clustering is less significant in these regions it still plays a role in the dynamics of the particles.

We have also shown that as a statistical representation of the ensemble, our method is extremely effective. The sedimentation rates of the Lagrangian particle simulations almost exactly match the sedimentation rates of the transfer operator predictions. Further, the regions of higher concentration at the bottom of the tank provided evidence that the clustering of the particles was accurately captured by the transfer operator. The regions of dense particle concentration very closely matched for both cases as can be seen in figure 5.17. Particles are strongly attracted to large scale structures in the flow. Such regions are prime candidates for accelerated aggregation processes such as rapid droplet growth in a cloud.

The obvious capabilities of the transfer operator approach to represent the dynamics of the inertial particles make it a good candidate for investigations into the influence of the particle phase on the fluid. It is possible to investigate how the energy spectrum of the turbulence is affected for a wide range of particle concentrations with no increase in the computational cost of the simulations.

However, the history force is clearly important for some regions of the parameters space. When the history force becomes significant, it is necessary to simulate the trajectories of the

individual particles rather than attempting to approximate their dynamics probabilistically using the transfer operator. The history force acts to suppress the clustering of the particles. In our simulations it destroys the clustering for a density ratio akin to sand in water. For the raindrop formation case the clustering is also reduced but to a lesser degree.

There is a clear transition in figure 5.18 between the region for which history is influential and the region for which it is not. For $St < 0.1$ the history force does not influence the clustering in the raindrop formation case. The same region defines the region for which the transfer operator successfully captures the clustering of active tracer particles. We conclude that the transfer operator approach can be safely used for $St < 0.1$.

CHAPTER 6

CONCLUSIONS

6.1 Summary

We have developed and tested novel modelling approaches for inertial particles, and explored in detail nuances of their dynamics. Through the first 3 chapters we constructed and verified our methods in steady paradigmatic vortical flows. In the fifth chapter we applied our methods in turbulent flows, showing that our Eulerian approach for inertial particles produces good results when compared with Lagrangian methods. Investigation of emergent ensemble behaviours, in both steady and turbulent flows, have allowed us to define particle behaviour in terms of simple understandable phenomena. The following is a summary of the specific works conducted in each chapter.

In the second chapter, we adapted existing numerical methods for integration of the Maxey-Riley equation. One approach for reducing the cost of numerical integration of the history force is to truncate the window of historical accelerations. However, doing so reduces accuracy significantly. The solution, as pointed out by Hinsberg et al., is to use an exponential kernel approximation for the tail of the history kernel [63]. Using the work of Hinsberg et al., we make an adaptation of their tail approximation which allows it to be used in higher order schemes. We showed that using the exponential kernel gives the same degree of accuracy as calculating the entire Basset integral, provided the window is sufficiently long. Further we point out that because the MR equation is a stiff equation, stability is improved by using an implicit method; this fact seems not to have been noted in the literature. Very small particle Stokes numbers make trajectories simulated with the explicit Adams-Bashforth method unstable. Although deviations from fluid velocities are limited for small Stokes, we showed that particles display inertial properties comfortably below the limit of instability for a step size of $t = 10^{-3}$. By using

an implicit method, such as the BDF2, problems of instability are avoided.

Our numerics compared well with an analytical solution of particles migrating radially under rigid body rotation. Comparison was also made with experimental data from Esperance et al.[3]. With confidence in our numerics, we used our method to investigate the influence of model choice on particle trajectories in a vortical flow. Specifically, we compared trajectories with and without history to ascertain when it should be used in later modelling. We found that although history was less influential for large density ratios, it still played a role for all densities.

In the third chapter, motivated by an Eulerian description of inertial particles, we discretised a transfer operator that acts on the probability density function of particle positions. Such an approach has computational complexity independent of the number of particles in the system. In order to further improve efficiency, we collapsed the dynamics of the particles onto an inertial manifold, the mathematical description of which was derived by Haller [4]. After reducing the dynamics of the system, sufficient resolution is achieved to effectively capture the important clustering properties of the particles. The method is compared against simulations containing large numbers of Lagrangian particle trajectories. Inertial behaviours of the particles are accurately maintained up to $S/R \approx 0.1$, at which point particle behaviours begin to change. Instead of being attracted to the separatrix dividing vortices, particles are attracted to a wider band of orbits. Our discretisation of the transfer is unable to capture these dynamics and so its usability is limited to values of $S/R < 0.1$.

In the fourth chapter, we compared novel collision detection algorithms with an existing benchmark approach. Collisions between particles are important in many natural processes, for example raindrop formation and planetary accretion [25, 69]. Therefore, it is important to provide methods for modelling collisions that can be used with our transfer operator approach. Modelling collisions with a discretised transfer operator has algorithmic complexity independent of the number of particles, compared to the benchmark we tested against, which has complexity of $O\left(N^{3/2}\right)$.

We also demonstrated that tracking the volumes surrounding trajectories is an effective way of detecting collisions. However, the said approach requires the calculation of the tangent space, which is only possible in prescribed analytical flows. For particles advected in a discrete velocity field, for example the result of a numerical solution of the Navier-Stokes equations, the tangent space cannot be calculated analytically. For this reason, volume tracking using the tangent space is only suitable in continuous analytically defined velocity fields.

When comparisons between collision locations are compared between the methods, excellent agreement is achieved. Remarkably, both approaches produced similar collision rates to the cell method in the example shown, although the quality of results obtained using the transfer operator is dependent on particle parameters and grid resolution. Whilst it requires further development, we are able to show that a collision detection algorithm with algorithmic complexity independent of the number of particles is achievable using our approach.

In the fifth chapter we used our methods in a decaying turbulent flow. Our discretisation of a transfer operator closely matched the clustering of particles in Lagrangian simulations and exactly matched particle sedimentation rates. Clustering is successfully captured when the Stokes number is less than 1, but does not produce reductions in clustering for further increases in St. We also investigated the clustering of particles under the influence of the history force, as there are discrepancies between findings from different authors [7, 31]. Interestingly, we found history to be more influential in our decaying turbulent flow than it had been in the forced turbulence tested by others. It appears that including history in modelling is particularly important in decaying turbulent flows.

6.2 Common themes

6.2.1 On the history force

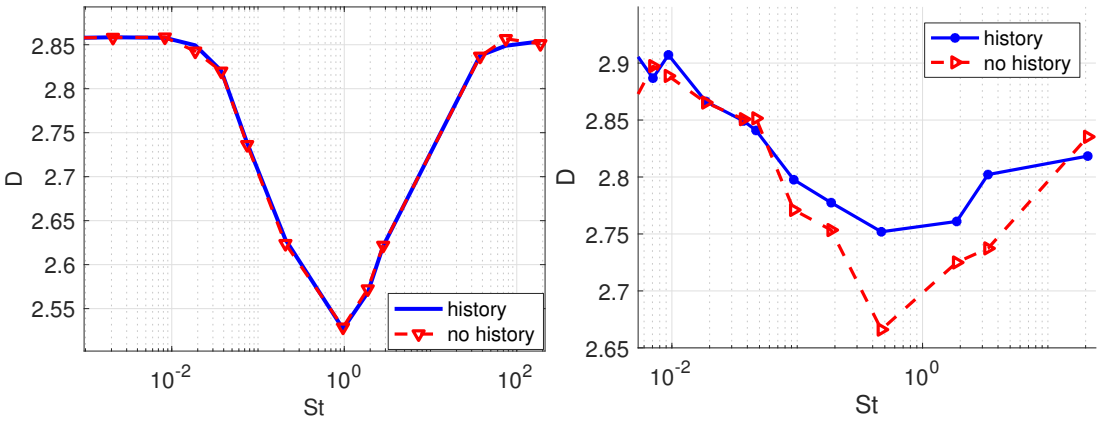
Throughout the thesis we check to see how the often neglected history force influences the specific dynamics relevant to the chapter. We found that history is most influential for small density ratios for all measured properties. The specific example we continually make reference to is sand particles in water, $\rho_p/\rho_f \approx 2$ or density parameter $R \approx 0.4$. For larger density ratios, and the same value of St, the history force has less of a bearing on particle dynamics. This theme is common for all properties we investigate: for low density ratios, history reduces the vortex ejection rate, path deviation, collision rates and clustering. Of course, all these properties are closely related and are all essentially behaviours that depend on vortex ejection. The radial migration rate controls how quickly particles reach attractive regions between vortices, which are exactly the regions in flows where particles cluster and are most likely to collide.

Using steady flows is supremely useful, because the relatively simple dynamics that particles display within them are easy to understand. Knowledge of particle behaviour in steady flows, opens channels of understanding towards particle behaviours in more complex flows. Clustering behaviour is similarly influenced by history in turbulent flows, as it is in steady vortical ones. It is considered in the literature that particles cluster in regions of low vorticity and high strain rate [18]. Such regions are analogous to the regions between vortical structures in the steady flows we investigate in the first three chapters.

In the fifth chapter, we chose to investigate the effect of history on the behaviour of particles in a decaying turbulent flow. Whilst there are recent studies on the behaviour of particles in forced turbulence in the presence of history, there are no recent studies to our knowledge that investigate inertial particles in a decaying flow. We find that historical effects on particle clustering are far more prevalent in a decaying flow. We attribute this to the difference in decay rates between the history kernel and the velocity of the fluid. Particle accelerations relative to the fluid decay much faster than the Basset kernel ($O(t^{-\frac{5}{2}})$, compared with, $O(t^{-\frac{1}{2}})$). Recent accelerations are much smaller than historical ones in a decaying flow. Therefore, the influence of the history force grows

relative to the other forces as the simulation progresses.

To ensure that our findings were unique to decaying flows, we have investigated clustering in a steady superposition of ABC flows. The superposition of flows was the same one used for the initial conditions of the decaying flow in the fifth chapter. Figure 6.1 shows how the correlation dimension D changes as a function of St with and without history. We find that the history force has a much reduced effect compared with its influence in decaying turbulence. The results are more closely aligned with the findings of Daitche [30].



(a) *water in air* $\rho_p/\rho_f \approx 1000$ ***(b)*** *sand in water* $\rho_p/\rho_f \approx 2$

Figure 6.1: *Using the initial condition of the decaying flows from the fifth chapter, we investigate particle clustering in a steady flow with multiple length scales, with and without history. History has no effect on clustering for the large density ratio case of water in air, but has a considerable effect on the clustering for the lower density ratio case. In both cases, the influence of history is significantly smaller than it was in the decaying flow. These results support Daitche's findings in forced turbulent flows [7].*

It appears that decisions about including history in an application should be based both on the value of particle parameters and on the form of the flow. A general rule is not easily obtained for the type of flow for which history will be most influential. What we can say with reasonable confidence is that in non-decaying flows, with large density ratios, history has little influence on vortex ejection, clustering, or particle collisions. Consequently, investigations into cloud microphysics are safely conducted without inclusion of the Basset force. Even in decaying flows, for water particles in air, history only begins to reduce particle clustering for $St > 0.1$.

Whilst shown to be accurate experimentally for small particle Reynolds numbers, the Basset force with a decay kernel of $t^{-1/2}$ cannot be used when the particle Reynolds number becomes too large. It is well known that for finite particle Reynolds numbers, the history kernel decays faster than the $O(t^{-\frac{1}{2}})$ version derived by Basset. Our results in decaying flows present a physical reason for why that must be the case. It is unphysical that disturbances caused by the particles should decay slower than the flow in which they reside. The reason for the difference is that

advective terms are ignored in Maxey and Riley's derivation. Even for the low particle Reynolds numbers we restrict ourselves to, vorticity transport is dominated by advection far from the particle. Advection is a much faster process than diffusion meaning that neglecting advection ignores the largest contribution to the decay of the flow. It is probable that the disturbance flow around a particle should decay at a similar rate to the overall flow, and thus the influence of historical accelerations should decay with the fluid. It is only when a decaying flow is used that the problems with the slowly decaying kernel become obvious. We hope that these results encourage work on numerical implementations of equations of motion for finite particle Reynolds numbers.

6.2.2 On parameters of particles

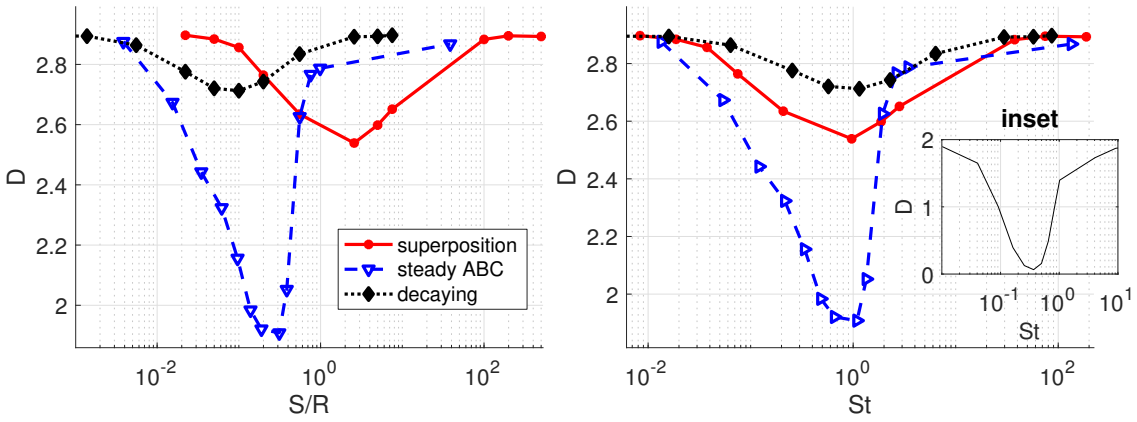
Clear throughout the thesis is the dependance of each of our methods on the St number. Our discretisation of a transfer operator, W_{ij}, has accuracy that is dependent on St remaining smaller than some threshold. Closely related is the degree of clustering shown by particles, which reaches a peak for $St \approx 1$. The influence of history also has a strong dependance on St, with its influence growing with increasing St until it reaches a maximum at $St \approx 1$.

An appropriate definition of the Stokes number for our different flows is not obvious. The Stokes number is hard to define because it is not always clear which timescale is most relevant to the particles. In steady flows, we defined T as $T = L/U$, where we chose L as the periodicity of the domain, and U as the root mean square velocity. In turbulent flows, T was defined as the Kolmogorov timescale, which is the timescale that characterises the smallest scales. We want to understand how the two compare from the perspective of the particles. Looking through the particle's eyes, fluid velocities change, both because the particle moves and because of the unsteady nature of the flow. The timescale, T, needs to reflect how quickly the flow changes from the perspective of the particle.

In this section, we will compare the results from the various chapters to define exactly how particle parameters should be normalised in our different flows based on particle behaviour. Let us use the general consensus that a Stokes number of 1 corresponds to the point where the particle response time, S/R, is equal to an appropriate fluid timescale T. When this happens, we see maximal particle clustering in turbulent flows, and maximal influence of the history force. The question is, what constitutes maximal clustering in the steady flows used in the earlier chapters?

In the third chapter, we explored in detail the large Stokes number behaviours of inertial particles that our transfer operator approach is unable to capture. We discovered that at some crucial parameter value, particles sufficiently ignore fluid streamlines such that they no longer organise along separatrices for long times. Let us define this phenomenon as analogous to peak clustering in turbulent flows. In the fifth chapter, we measured clustering using the well known correlation dimension, D. If we use the same measure to analyse clustering in steady flows, we

will be able to compare the results and identify an appropriate T for our steady flows. In figure 6.2 we have plotted the correlation dimension as a function of both S/R and $St = S/(RT)$. We find that for the steady 3-dimensional flows, an appropriate choice for the size of T is the reciprocal of the root mean square velocity $1/U$. Therefore, it appears that 1 is the appropriate length scale in our steady flows. In the turbulent flow, as before, an appropriate length scale is the Kolmogorov length scale η. In the 2-dimensional flow a different story emerges, and we find that, when L is set to 1, peak clustering occurs for $St \approx 0.4$.



(a) *As a function of S/R*

(b) *As a function of St*

Figure 6.2: *When the correlation dimension is plotted as a function of Stokes, particle clustering reaches a maximum at $St \approx 1$. This is true of a single ABC flow (dashed line), a superposition of ABC flows (solid line), and a decaying turbulent flow (dotted line).* ***Inset****: The correlation dimension is plotted as function of Stokes in the 2-dimensional flow used in chapters 1, 2 and 3. Peak clustering occurs at $St \approx 0.4$.*

Finally, we can link all of the results through a single parameter $St = S/(RT)$, which can be used to predict the peak influence of history and particle clustering, both in steady flows and turbulent ones. Whilst peak clustering did not occur at exactly $St = 1$ for all our flows, it is a reasonable estimate in all cases. Where a reasonable estimate can be made about the value of St which will produce peak clustering, then the limit of validity of the transfer operator can also be reasonably determined.

6.2.3 On model choice

We now review appropriate model choice with reference to particle behaviour. It makes sense to split the discussion between applications where history is required, and applications where it is not. Let us assume that the particle Reynolds number is sufficiently small to allow the MR equation to be used. In this case, where particle density ratios are large, history can be safely neglected. Where density ratios are smaller, history is important to particle dynamics provided

the Stokes number is above a certain value. The appropriate value of Stokes is dependent on the flow, but a reasonable prediction of the importance of history can be made using figure 2.7.

Where possible, it is advantageous to use the discretisation of the transfer operator to model inertial particles. An Eulerian representation of particle dynamics allows phenomena such as two-way coupling between the particle and fluid phase to be modelled. It is also efficient at modelling large ensembles, because increasing the number of represented particles has no associated performance penalty. Such efficiency makes a transfer operator approach an excellent candidate for modelling collisions, which is computationally expensive for Lagrangian particles. Another advantage of the transfer operator is that, in steady flows, the long term state of the system is given by the eigenvectors of W_{ij}. Eigenvectors can be calculated with power iteration algorithms allowing the longterm behaviours to be calculated efficiently.

Our discretisation of the transfer operator may only be used in situations where the influence of the history force is negligible. Further, because our discretisation of the transfer operator relies on the inertial equation, its accuracy reduces with increasing St. For $St \approx 1$, even qualitative accuracy is lost due to the fundamental changes in particle dynamics discussed in the previous section.

In order to optimise numerical efficiency and accuracy, we believe that a multi-method modelling approach should be considered for applications with a range of particle sizes. An ideal model would combine the methods we have developed to efficiently model processes with a broad range of parameter regimes. In instances where there are large numbers of small particles, the transfer operator approach should be used. As the particle diameter increases in an aggregation process, such as planetary formation or raindrop growth, the larger particles should be modelled as Lagrangian particles using the full MR equation. At some point, when particle diameters exceed several times the Kolmogorov length scale, immersed boundary methods should be considered which can account for the complex effects of finite particle size on the fluid velocity field.

For the scales we aimed to represent, our hierarchy of approaches satisfactorily covers a wide particle parameter space. We have opened the door to investigations of two-way coupling between small particles and turbulent fluid flows, and identified instances where historical effects are particularly pronounced. Whilst we primarily investigated density ratios representative of water in air and sand in water, our methods can be used for any density ratio. The tools we have provided allow significant further understanding of inertial particles to be developed.

6.3 Further work

Whilst we have explored a number of avenues, and developed several new methods, problems remain unsolved. Further, the code produced during this PhD contains a number of additional features that have not been fully explored. These can readily be exploited to answer scientific

questions. In the following, we detail the areas that are ripe for further research.

6.3.1 The Basset history force

In the fifth chapter, we showed that the Basset history force was particularly significant in decaying turbulent flows. Our results are different to those conducted by other researchers in forced turbulent flows. In both cases, comparisons with experimental data would help to verify the findings of the numerical studies, and to improve understanding of the physical mechanisms of particle clustering.

Unfortunately, there is limited experimental data with which comparison can be satisfactorily made. Difficulties in experiment arise for a number of reasons. Firstly, the conditions in simulations that are most suited to investigating the influence of history on clustering, do not coincide with conditions that are readily recreated in the laboratory. To explore specifically the influence of inertia, clustering is often investigated numerically in zero gravity. Of course, gravity is always present in real laboratory experiments. Gravitational effects have been shown to enhance clustering for $St > 1$ whilst inhibiting it for $St < 1$ [104]. Therefore, investigations specifically concerned with inertial effects on particle clustering, will produce different results to those that include gravity. Another difficulty is maintaining consistent particle concentration. Sedimentation must be countered by consistent addition of further particles. Finally, accurately measuring the positions of particles in 3-dimensions in a tank is largely limited to planar views.

Despite the difficulties, some experimental datasets exist for particle clustering in forced turbulence. Salazar et al. investigated particle clustering in homogenous isotropic turbulence experimentally. Results were compared with numerical simulations of Lagrangian particles, but both the pressure term and history term were neglected [105]. Interestingly, Salazar et al. found that the clustering was under predicted in the simulation. In their experiments, particles were many times denser than the fluid with $\rho_p/\rho_f = 1375$. For such a large density ratio and a forced turbulent flow, the influence of history is expected to be negligible.

Very recently, an experiment was conducted by Petersen et al. [106], which measured particle clustering for a range of density ratios, in forced, homogenous isotropic turbulence. History is more likely to influence particle clustering when density ratios are smaller. Consequently, comparison between their results and numerical simulations of particles with history would be valuable. Unfortunately, their paper was published on the 14 February 2019, leaving insufficient time to make any sort of meaningful comparisons.

Experimental data investigating particle clustering in decaying turbulence appears to be missing in the literature. Such work would be valuable for comparison with our numerical results, although difficult to measure due to the fast decay rate of unforced flows.

Finally, numerical schemes for forms of the Basset-Bousinesq-Oseen equation valid for finite Particle Reynolds number are sorely needed. See for example Lovalenti and Brady [53], whose equation has a faster decaying history Kernel. Their history kernel decays faster than the

Basset kernel, because advective terms were partially considered in their derivation. Rather than deriving an equation of motion for particles in a Stokes flow, Lovalenti and Brady used the Oseen flow. A faster decaying history kernel is expected to reduce influence in decaying turbulence significantly.

6.3.2 Two-way coupling

A key motivation for the transfer operator approach is that it allows the influence of the particle phase on the fluid to be explored. It has been found that for high concentrations of particles, the particle phase influences the energy dissipation rate of the flow [21]. Crucially, our transfer operator allows the mechanisms of two-way coupling between small particles and the fluid phase to be explored through buoyancy forcing and momentum transfer, as is considered in studies such as [107]. Under the approximation that the particle phase can be treated as a fluid of another density, MOBILE allows consideration of the buoyancy force caused by regions of different density fluid. Further work is required to include the momentum transfer between the particle and fluid phase, which can be incorporated by considering that the force imposed on the fluid is equal and opposite to the force imposed on the particles.

With these methods implemented, we would like to explore the influence of the particles on the energy spectrum of decaying turbulence and answer questions about the influence of two-way coupling on the clustering properties of the particles. Does fluid particle feedback increase collision rates in turbulent flows or reduce them? Can specific particle topologies enhance turbulent energy whilst others suppress it? For example, if particles appear in dense clusters with some spatial frequency, are those spatial frequencies excited in the fluid phase?

6.3.3 Improving the transfer operator method

The accuracy of our discretisation of a transfer operator for inertial particles is dependant on St. For $St > 1$ the inertial manifold is not a good qualitative prediction of particle velocities. The number of attractive particle paths increases such that the separatrices are no longer the only likely long term location of particles. However, where the attractive paths remain unique, particle velocities occupy a predictable region of the space close to the manifold. For this parameter regime, the transfer operator approach could be improved by including a small region of the particle phase space either side of the manifold. By changing the coordinate system so that the active tracer system is given by,

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{w} + \mathbf{u}(x,t) \\ \ddot{\mathbf{x}} &= \frac{3}{2} R \frac{D\mathbf{u}(\mathbf{x},t)}{Dt} - \frac{R}{St}\mathbf{w}, \end{aligned} \tag{6.1}$$

where $\mathbf{w} = \mathbf{v}(t) - \mathbf{u}(\mathbf{x}, t)$, rather than,

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{v} \\ \ddot{\mathbf{x}} &= \frac{3}{2} R \frac{D\mathbf{u}(\mathbf{x}, t)}{Dt} - R/St(\mathbf{v}(t) - \mathbf{u}(\mathbf{x}, t)), \end{aligned} \tag{6.2}$$

a transfer operator could be defined that acts on the phase space $\mathbf{x}(t) = [\mathbf{w}(t), \mathbf{y}(t)]$. In this alternative phase space, the volume requiring discretisation is dramatically reduced. With only a small region of the phase space required, the computational requirements of a higher dimensional transfer operator would be tractable.

It is also possible to improve the remapping algorithm. The remapping method introduced in chapter 3 only takes into account the translation of the cell centroid. In figure 3.23, we demonstrated that certain flow features produce erroneous results when only the translation of the cell centroid is taken into account. For example, a saddle in the particle velocity field is problematic because the cell centroid can remain stationary, whilst particles move around it. Similar problems occur where a cell perfectly encompasses a symmetrical vortex. Particle trajectories circle the vortex centre whilst the centroid remains stationary. By taking into account the relative motion of the corners of cells, the remapping algorithm could be improved significantly. Considering corner translation alleviates the problems of symmetrical movement of particles either side of their centroid.

Despite the simplicity of our approach, remarkable agreement was achieved between Lagrangian and Eulerian simulations, which encourages further improvement of the method.

A discretisation of a transfer operator that can successfully capture the inertial properties of particles is a useful tool, and one we intend to further develop to include as many physical properties of particles as possible.

APPENDIX A

COMBINING THE TRANSFER OPERATOR AND MAXEY-RILEY SOLVER WITH MOBILE

In the fifth chapter, the discretisation of the transfer operator is investigated in decaying turbulent flows using the Navier-Stokes solver MOBILE. A significant library of functions was created in this PhD to provide a versatile inertial particle toolbox. In this section a short user guide is presented for the code.

A.1 Capabilities

A suite of functions has been produced that provides a number of simulation options. These are primarily split into two categories: simulations using the discretisation of the transfer operator and simulations of Lagrangian particles. For both simulation types it is possible to simulate in 2 or 3 dimensions with or without gravity for two different test cases: a turbulent jet and a decaying superposition of ABC flows. Other initial conditions can be straightforwardly implemented through MOBILE, provided they satisfy the existing boundary conditions.

The code is massively parallel using the open MPI interface for communicating between processes. Parallelisation is handled by splitting the domain between processes. When particles pass from one section of the domain to another, they are passed using open MPI's `MPI_Irecv` and `MPI_Isend` functions. Boundary conditions are passed similarly when the transfer operator is used.

For the Lagrangian particle simulations, trajectories can be calculated to second order accuracy with or without history. A number of options also exist for collision simulations. Particle collisions can be detected using the cell method detailed in chapter 4. Options for the nucleation rate, collision type, and collision threshold are provided in `input_collisions.dat`. For the trans-

fer operator, density forcing on the fluid phase has also been implemented, although not explored in this thesis. Density forcing can be switched on by adding the line, `MRadvect:=assign(ON);` to the MOBILE input file, `input_setup.dat`.

A.2 Code structure

The particle toolbox has a separate memory allocation and function library to MOBILE. The aim was to limit interaction between the phases as much as possible so that the code could be used with any Navier-Stokes solver. Two intermediary functions are used to communicate between the fluid and particle phase: `MR_init_mobile` gets the domain size, allocates memory and applies initial conditions; `MR_update_mobile` gets the new velocity fields and calls all updating functions. Figure A.1 shows the order of the key operations performed by the code within one time step. Communication with MOBILE is indicated by the red dashed boxes.

The majority of function calls come from `MR_update_mobile`, which conducts operations in the following order: gets new velocity field from MOBILE; passes particles between processes; advects particles using the solver detailed in the second chapter; outputs data to file.

A.3 Input files

Three input files are used for the particle phase: `input_MR.dat`, `MRflowvariables.dat` and `input_collisions.dat`. Each is presented in a table along with a brief description of each input.

The input file `input_MR.dat`, contains the majority of inputs and is presented in table A.1. Capabilities that are specific to either the Lagrangian or Eulerian descriptions of particles are indicated. Where an input has a number of possible options they are displayed in table A.2. Each option corresponds to a different input.

An input file is also required to define the initial flow parameters. The flow parameters for an initial superposition of ABC flows is contained within `MRflowVariables.dat`, and is presented in table A.3.

To investigate aggregation processes, such as raindrop formation, an additional input file is required. Choices can be made about collision types and particle nucleation rates in the file `input_collisions.dat`. Each of the inputs is presented in table A.4.

A.4 Output files

Output files can be produced in various formats. For the transfer operator, PNG images of planar slices through the domain can be produced, as well as a full print out of the PDF of particle positions to a CSV file. For the Lagrangian particle simulations, particle positions and velocities can be printed to CSV files. When collisions are detected, collision locations, pairs and particle statistics can also be produced. Each case is outlined briefly below.

Parameter	Default value	Description	Specific to?
N	20000	number of particles	Lagrangian
dt	0.001	time step of MR equation solver	
pf	1000	fluid density	
pp	2000	particle density	
v	0.000010	dynamic fluid viscosity	
diam	0.0010	particle diameter	
histSwitch	0	turns on the history force	Lagrangian
PerronSwitch	1	turns on transfer operator.	Eulerian
restartSwitch	0	restarts a simulations based on the latest output files in the folder	
collisionsSwitch	0	turns on particle collision detection	Lagrangian
imFreq	0.5	frequency of output for particle positions, PDF and images	
binFreq	10.0	frequency of output of full domain data	
imageSwitch	1	turns on image generation	Eulerian
nPlanes	3	number of image planes	Eulerian
minP	0.0	colour saturation minimum as a factor of mean concentration	Eulerian
maxP	2.0	colour saturation maximum as a factor of the mean concentration	Eulerian
plane	0 1 2	orientation of normal to the image plane ($x = 1$ $y = 2$ $z = 3$)	Eulerian
colScheme	4 4 4	colour scheme: for options see table A.2	Eulerian
plnHght	0.5 0.5 0.4	position of image plane with orientation given by 'plane'.	Eulerian
initPartDens	10000000	Particle number density, N/m^3	Eulerian
ParticlePert	1	initial distribution of particles: for options see table A.2	Eulerian
initTime	0.0	time at which particle phase should be inserted.	
printVels	0	should the velocity field be printed to file?	

Table A.1: *The main input file for the particle phase is* `input_MR.dat`*. Some inputs are specific for Lagrangian particle simulations, and some are for Eulerian simulations, in both cases this is indicated in the table. The format for each input is '*$variablename = value$*' and they must appear in the file in this order. For inputs which turn on or off a capability, e.g 'printVels', off*$= 0$ *and on*$= 1$.

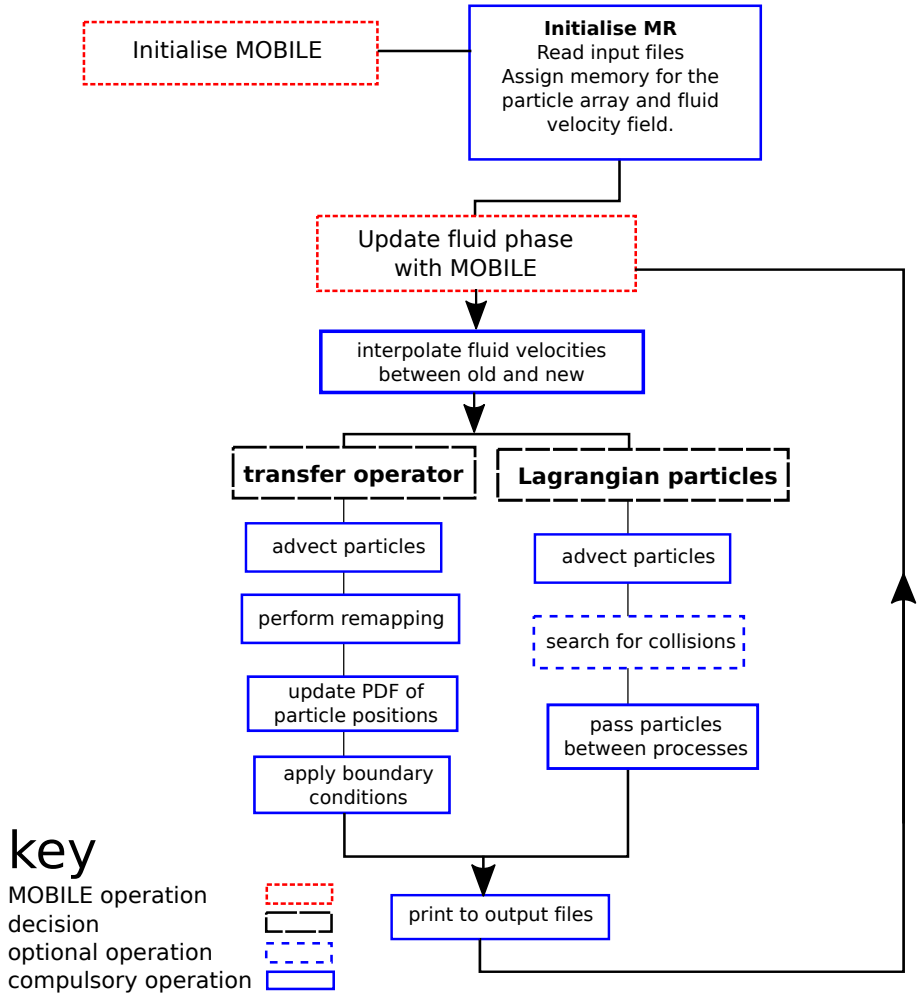


Figure A.1: *The Maxey-Riley solver and transfer operator code is kept as separate as possible from MOBILE. Interactions occur once a time step when the new fluid velocities are passed to the Maxey-Riley solver.*

A.4.1 PNG images

Each PNG image is produced for a specific plane orientation, position and time. Separate images are produced for each process, which can then be stitched back together. The filename convention for PNG images is based on the orientation plane, the process that produced the file, and the output time. This combination produces a filename given by "`MassDensity<orientation plane: x = 0, y = 1, z = 2>Plane<process>Proc<time/imFreq>.png`", which looks confusing but will be made more explicit with an example. Consider an image plane oriented with the vertical 'z' axis, produced by the main process, process 0. The image is produced after two seconds and the image frequency, `imFreq`, is set to 1. The resulting file name is "`MassDensity2Plane0Proc2.png`". A number of different colour schemes are implemented and can be chosen using the colScheme pa-

colScheme	0 =GREY, 1 =GREEN, 2 =BLUE, 3 =RED, 4 =COLOUR 10 =PINK
ParticlePert	1 =HOMOGENEOUS, 2 =JET and 3 =STRIPES
plane	0 = x, 1 = y, 2 = z

Table A.2: *Set of options for the colour scheme, initial spatial distribution of particles, and the orientation of the normal to the image plane.*

Parameter	Default value	Description
flowConsts	0.09 0.1 0.11	flow constants A, B and C.
kmax	14	number of superpositions

Table A.3: *`MRflowVariables.dat` provides the flow constants for a superposition of ABC flows.*

Parameter	Default value	Desciption
nucleate	1	Turn on particle nucleation.
relaxationFactor	1	The factor by which the collision threshold should be relaxed.
coalesce	1	Turn on coalescing particles. When set to 0 colloidal particles are simply removed from the simulation.
rainThreshold	0.002	Particles of greater diameter than this are removed from the simulation as rain.
nucleationRate	0.001	Maximum nucleation rate as a proportion of the total number of particles.

Table A.4: *`input_collisions.dat` provides options for collision simulations.*

rameter in `input_MR.dat`. The options are: 0 =GREY, 1 =GREEN, 2 =BLUE, 3 =RED, 4 =COLOUR 10 =PINK and 11 =SAND. An example image is shown in figure A.2.

A.4.2 Particle positions

For the Lagrangian particle simulations, output files contain particle positions and velocities. Each file is a CSV file which follows a similar naming convention to the PNG images. Each filename is of the form: ‘‘`Particle<process>Positions<time/imFreq>.csv"`. As an example, the name of a particle position file produced by process 1 after 2 seconds with `imFreq` set to 0.5 would be ‘‘`Particle1Positions4.csv"`. For a space defined by $\mathbf{x} = [x, y, z]$ and particle velocity given by $\mathbf{v} = [u, v, w]$, the format of files storing particle positions and velocities is given in table A.5.

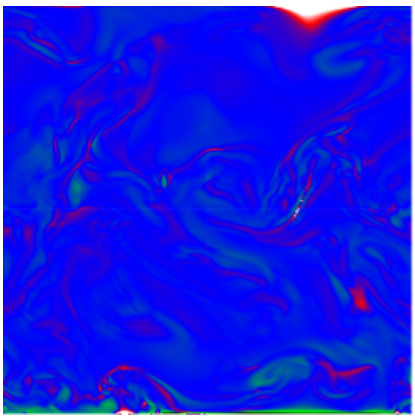


Figure A.2: *A planar slices in the x-z plane using the COLOUR scheme. The image shows regions with high and low particle concentration in the early stages of a sedimentation simulation.*

time	x	u	y	v	z	w	label	original process	current process
#	#	#	#	#	#	#	#	#	#

Table A.5: *Particle positions and velocities are printed at regular intervals along with the time, particle label, original process and current process.*

x	y	z	time
#	#	#	#

Table A.6: *Collision locations and times are given in* `collisionLocation.csv`.

A.4.3 Collision statistics

When collisions are considered, a user may be concerned with collision locations, times and collision pairings. As well as the distributions of the particle sizes. Outputs from collision simulations come in three files: `collisionLocation.csv`, `collisionRecord.csv` and `particle<process>Statistics<time/imFreq>.csv`. Collision locations are printed in `collisionLocation.csv` in the format shown in table A.6.

Collision pairs are given in `collisionRecord.csv`. Each particle has a unique label attached to it allowing collision pairs to be tracked. For a collision involving particle A and particle B, the format of the collision record is presented in table A.7.

Statistics for the whole ensemble are printed in `particle<process>Statisitics<time/imFreq>.csv`, the format of which is presented in table A.8. Particle size distributions can be tracked through time using this file.

label A	diameter A	label B	diameter B	time
#	#	#	#	#

Table A.7: *Collision labels, times and diameters are given in* `collisionRecord.csv`*. Each particle has a unique label attached to it so that each can be kept track of.*

label	diameter	number of collisions	elimination switch
#	#	#	#

Table A.8: *Statistics of all particles are printed into files of this format at regular intervals. Number of collisions indicates the number of collisions a particle has been involved in. Elimination switch indicates whether or not a particle has been removed from the system. Particles are removed from the system after a collision if the user has chosen not to allow particles to coalesce. If particles are allowed to coalesce, then the smaller of the two particles involved in the collision is absorbed into the larger and the smaller particle is eliminated from the system.*

APPENDIX B

TESTING THE CALCULATION OF THE ENERGY SPECTRUM AND THE INTEGRAL LENGTH SCALE.

In the fifth chapter we calculated the energy spectrum of the turbulent flows used. In order to test the calculation of the integral length scale L and the one-dimensional energy spectrum $E(||\mathbf{k}||, t)$, a test flow with a known length scale and dominant frequency is used. This is given by,

$$\mathbf{u}(x,0) = \begin{bmatrix} B\cos(k_0 z) + C\sin(k_0 y) \\ C\cos(k_0 x) + A\sin(k_0 z) \\ A\cos(k_0 y) + B\sin(k_0 x) \end{bmatrix}, \quad k_0 \quad = 10, \tag{B.1}$$

where $A = 0.09$, $B = 0.1$, $C = 0.11$. The flow presented in equation B.1 has one dominant frequency at $||\mathbf{k}|| = 10$ and a corresponding integral length scale that should be equal to $2\pi/k_0 = 0.6283$.

Figure B.1 shows the energy spectrum for the velocity field given in equation B.1. The peak is very close to the expected position of $||\mathbf{k}|| = 10$. The integral length scale gives $L = \frac{G\sum_k k^{-}1E(k,t)}{\sum_k E(k,t)} = 0.6133$, where G is the length of one side of the domain.

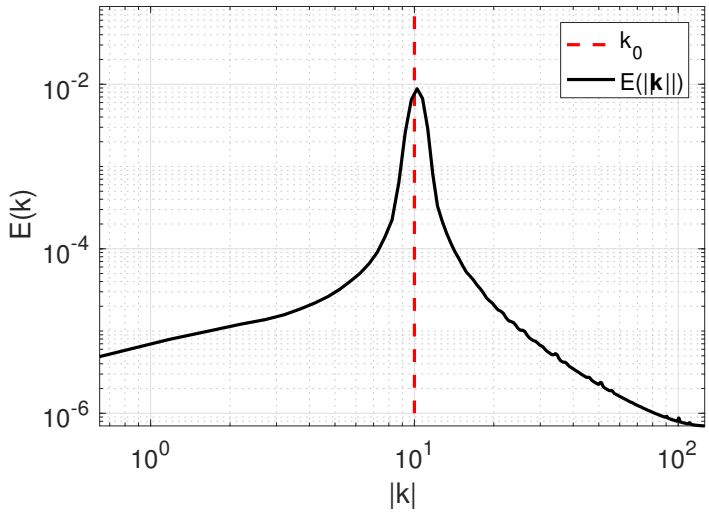


Figure B.1: *We test our calculation of the energy spectrum $E(||\mathbf{k}||, 0)$ using the flow defined in equation B.1. The value of k_0 is indicated by the red dashed line. The integral length scale and energy spectrum are as expected for the test flow.*

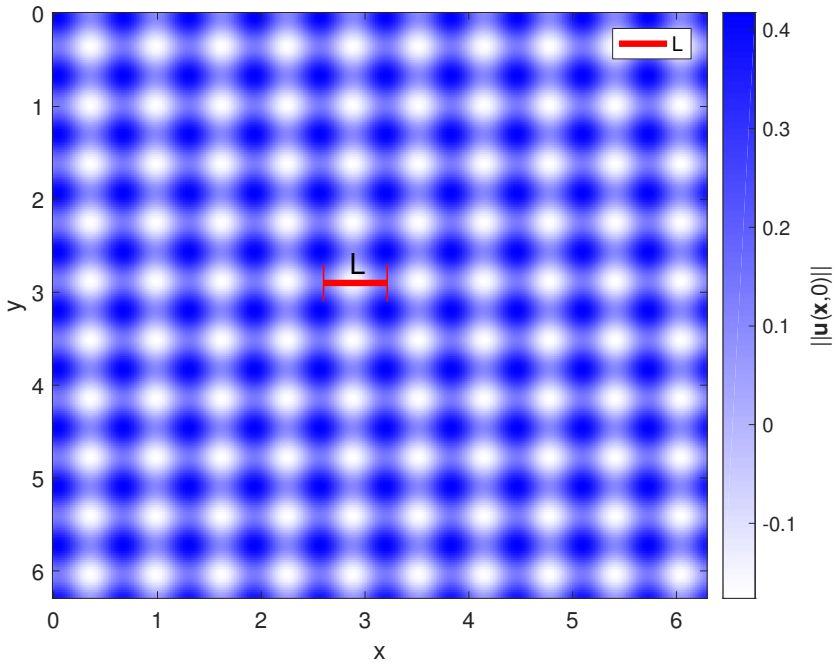


Figure B.2: *The magnitude of the velocity field, $||\mathbf{u}(x, 0)||$ is plotted in the plane $z = \pi$. The integral length scale $L = 0.6133$ is shown to be the distance between the centres of the regular vortical structures present in the flow.*

BIBLIOGRAPHY

[1] S. L. Jeffery S. Marshall, *Adhesive Particle Flow*.
Cambridge University Press, 2014.

[2] Taneda, "Experimental investigation of the wake behind a sphere at low reynolds numbers.," *Journal of the Physical Society of Japan*, vol. 11, no. 10, pp. 1104–1118, 1956.

[3] D. L'Espérance, C. F. M. Coimbra, J. D. Trolinger, and R. H. Rangel, "Experimental verification of fractional history effects on the viscous dynamics of small spherical particles," *Experiments in Fluids*, vol. 38, pp. 112–116, 2005.

[4] G. Haller and T. Sapsis, "Where do inertial particles go in fluid flows?," *Physica D*, vol. 237, pp. 573–583, 2008.

[5] J. Philip, "The probability distribution of the distance between two random points in a box."

[6] J. C. Vassilicos, "Dissipation in turbulent flows," *Annual Review of Fluid Mechanics*, vol. 47, pp. 95–114, 2015.

[7] A. Daitche, "On the role of the history force for inertial particles in turbulence," *Journal of Fluid Mechanics*, vol. 782, pp. 567–593, October 2015.

[8] Guardian, "Volcanic ash cloud cost European business up to €2.5bn, says EU," April 2010.

[9] H. N. Webster, D. J. Thomson, B. T. Johnson, I. P. C. Heard, K. Turnbull, F. Marenco, N. I. Kristiansen, J. Dorsey, A. Minikin, B. Weinzierl, U. Schumann, R. S. J. Sparks, S. C. Loughlin, M. C. Hort, S. J. Leadbetter, B. J. Devenish, A. J. Manning, C. S. Witham, J. M. Haywood, and B. W. Golding, "Operational prediction of ash concentrations in the distal volcanic cloud from the 2010 Eyjafjallajökull eruption," *Journal of Geophysical Research*, January 2012.

[10] A. Folch, A. Costa, and S. Basart, "Validation of the FALL3D ash dispersion model using observations of the 2010 Eyjafjallajökull volcanic ash clouds," *Atmospheric Environment*, vol. 48, pp. 165–183, March 2012.

[11] V. Matthias, A. Aulinger, J. Bieser, J. Cuesta, B. Geyer, B. Langmann, I. Serikov, I. Mattis, A. Minikin, L. Mona, M. Quante, U. Schumann, and B. Weinzierl, "The ash dispersion

over Europe during the Eyjafjallajökull eruption – comparison of CMAQ simulations to remote sensing and air-borne in-situ observations," *Atmospheric Environment*, vol. 48, pp. 184–194, March 2012.

[12] W. H. O. (WHO), "Burden of disease from household air pollution for 2012," tech. rep., World Health Organization (WHO), World Health Organization, 1211 Geneva 27, Switzerland, March 2014.

[13] W. H. O. (WHO) and the United Nations Children's Fund (UNICEF), "Progress on drinking water, sanitation and hygiene: 2017 update and SDG baselines.," tech. rep., World Health Organization (WHO) and the United Nations Children's Fund (UNICEF)., Geneva, 2017.

[14] K. Saralidze, L. H. Koole, and M. L. Knetsh, "Polymeric microspheres for medical applications," *Materials*, vol. 3, pp. 3537–3564, 2010.

[15] G. Falkovich and A. Pumir, "Sling effect in collisions of water droplets in turbulent clouds," *American Meteorological Society*, December 2007.

[16] M. V. Gregory Falkovich, Mikhail G. Stepanov, "Rain initiation time in turbulent warm clouds," *Journal of Applied Meteorology and Climatology*, vol. 45, pp. 591–599, 2006.

[17] J. T. Kelly, C. C. Chuang, and A. Wexler, "Influence of dust composition on cloud droplet formation.," *Atmospheric Environment*, vol. 41, pp. 2904–2916, 2007.

[18] K. D. Squires and J. K. Eaton, "Particle response and turbulence modification in isotropic turbulence," *Physics of Fluids A: Fluid Dynamics*, vol. 2, 1990.

[19] R. A. Shaw, "Particle-turbulence interactions in atmospheric clouds," *Annual Review of Fluid Mechanics*, vol. 35, no. 1, pp. 183–227, 2003.

[20] C. Poelma and G. Ooms, "Particle-turbulence interaction in a homogeneous, isotropic turbulent suspension," *Applied Mechanics Reviews*, vol. 59, March 2006.

[21] S. Elghobashi and G. C. Truesdell, "On the two-way interaction between homogeneous turbulence and dispersed solid particles. i: Turbulence modification," *Physics of Fluids A: Fluid Dynamics*, vol. 5, no. 1790, 1993.

[22] S. Elghobashi, "On predicting particle-laden turbulent flows," *Applied Scientific Research*, vol. 52, pp. 309–329, 1994.

[23] A. Ferrante and S. Elghobashi, "On the physical mechanisms of two-way coupling in particle-laden isotropic turbulence," *Physics of Fluids*, vol. 15, no. 315, 2003.

[24] F. Lucci, A. Ferrante, and S. Elghobashi., "Modulation of isotropic turbulence by particles of Taylor length-scale size.," *Journal of Fluid Mechanics*, vol. 650, pp. 5–55., March 2010.

[25] G. Falkovich, A. Fouxon, and M. G. Stepanov, "Acceleration of rain initiation by cloud turbulence," *Nature*, vol. 419, pp. 151–154, 2002.

[26] P. Davidson, *Turbulence: An introduction for scientists and engineers*. Oxford, 2nd ed., 2015.

[27] M. Uhlmann, "Interface-resolved direct numerical simulation of vertical particulate channel flow in the turbulent regime," *Physics of Fluids*, vol. 20, 2008.

[28] F. B. Usabiaga, R. Delgado-Buscalioni, B. E. Griffith, and A. Donev, "Inertial coupling method for particles in an incompressible fluctuating fluid," *Computational Methods Applied Mechanical Engineering*, vol. 269, pp. 139–172, 2014.

[29] H. R. Pruppacher and J. D. Klett, *Microphysics of Clouds and Precipitation.*, vol. 18. Springer, 2nd ed., 1997.

[30] A. Daitche, "Advection of inertial particles in the presence of the history force: Higher order numerical schemes," *Computational Physics*, Oct 2012.

[31] S. Olivieri, F. Picano, G. Sardina, D. Iudicone, and L. Brandt, "The effect of the Basset history force on particle clustering in homogeneous and isotropic turbulence," *Physics of Fluids*, vol. 26, 2014.

[32] G. P. Langlois, M. Farazmand, and G. Haller., "Asymptotic dynamics of inertial particles with memory," *Journal of Nonlinear Science*, vol. 25, pp. 1225–1255, 2015.

[33] P. G. Saffman, "The lift on a small sphere in a slow shear flow.," *Journal of Fluid Mechanics*, vol. 22, pp. 385–400, 1965.

[34] G. Magnus, "U ber die abweichung der geschosse," *Abhandlungen der Königlichen Akademie der Wissenschaften zu Berlin*, pp. 1–23, 1852.

[35] M. Provansal, L. Schouveiler, and T. Leweke, "From the double vortex street behind a cylinder to the wake of a sphere," *European Journal of Mechanics B/Fluids*, vol. 23, pp. 65–80, July 2004.

[36] B. Kallemov, A. P. S. Bhalla, B. E. Griffith, and A. Donev, "An immersed boundary method for rigid bodies," *Communications in Applied Mathematics and Computer Science*, vol. 11, pp. 79–141, 2016.

[37] M. B. Liu and G. R. Liu, "Smoothed particle hydrodynamics (SPH): an overview and recent developments," *Archives of Computational Methods in Engineering*, vol. 17, pp. 25–76, 2010.

[38] J. M. Burgers, *2nd Report on Viscosity and Plasticity*.
Amsterdam: North-Holland Publishing Company, 1938.

[39] S. A. Poison, "Memoire sur les mouvemens simultanes d'un pendule et de l'air environnant.," *Memoire de I' Academie des Science*, vol. 9, pp. 521–523, 1831.

[40] G. Green, "V. researches on the vibration of pendulums in fluid media," *Earth and Environmental Science Transactions of The Royal Society of Edinburgh*, vol. 13, pp. 54–62, 1835.

[41] G. G. Stokes, "On the effect of the internal friction of fluids on the motion of pendulums," *Transactions of the Cambridge Philosophical Society*, vol. 9, p. 8, 1851.

[42] Boussinesq, "Sur la resistance qu' oppose un liquide indetini en repos," *Comptes rendus de l'Académie des Sciences*, vol. 100, pp. 935–937, 1885.

[43] A. B. Basset, *A Treatise on Hydrodynamics With Numerous Tramples Volume 2*.
Cambridge [Eng.] Bell and co., 1888.

[44] C. W. Oseen, "Über die stokes'sche formel und über eine verwandte aufgabe in der hydrodymik," *Arkiv för matematik, astronomi och fysik*, vol. 6, no. 29, 1910.

[45] C. W. Oseen, "Über den gültigkeitsbereich der stokesschen wider- standsforme," *Arkiv för matematik, astronomi och fysik*, vol. 9, no. 19, 1913.

[46] E. E. Michaelides, "Review - the transient equation of motion for particles, bubbles and droplets," *Journal of Fluids Engineering*, vol. 119, June 1997.

[47] M. R. Maxey and J. J. Riley, "Equation of motion for a small rigid sphere in a nonuniform flow," *Physics of Fluids*, 1983.

[48] G. B. Jeffery, "Ellipsoidal particles immersed in a viscous fluid," *Proceedings of the Royal Society*, November 1922.

[49] E. Loth and A. J. Dorgan, "An equation of motion for particles of finite Reynolds number and size," *Environmental fluid mechanics*, pp. 187–206, 2009.

[50] P. M. Lovalenti and J. F. Brady, "The force on a bubble, drop, or particle in arbitrary time-dependent motion at small Reynolds number," *Physics of Fluids A*, vol. 5, September 1993.

[51] V. Galindo and G. Gerbeth, "A note on the force on an accelerating spherical drop at low-Reynolds number," *Physics of Fluids A: Fluid Dynamics*, vol. 5, 1993.

[52] E. E. Michaelides and Z. G. Feng, "The equation of motion of a small viscous sphere in an unsteady flow with interface slip," *International Journal of Multiphase Flow*, vol. 21, pp. 315–321, April 1995.

[53] P. M. Lovalenti and J. F. Brady, "The hydrodynamic force on a rigid particle undergoing arbitrary time-dependent motion at small Reynolds number," *Journal of Fluid Mechanics*, vol. 256, pp. 561–605, 1993.

[54] R. Mei and R. J. Adrian, "Flow past a sphere with an oscillation in the free-stream velocity and unsteady drag at finite Reynolds number," *J. Fluid Mechanics*, vol. 237, pp. 323–341, 1992.

[55] P. M. Lovalenti and J. F. Brady, "The temporal behaviour of the hydrodynamic force on a body in response to an abrupt change in velocity at small but finite Reynolds number," *Journal of Fluid Mechanics*, vol. 293, pp. 35–46, February 1995.

[56] M. R. Maxey, E. J. Chang, and L.-P. Wang, "Interactions of particles and microbubbles with turbulence," *Experimental Thermal and Fluid Science*, pp. 417–425, 1996.

[57] M. Abbad and M. Souhar, "Experimental investigation on the history force acting on oscillating fluid spheres at low reynolds number," *Physics of Fluids*, vol. 16, 2004.

[58] J. Weinstein, D. R. Kassoy, and M. Bell, "Experimental study of oscillatory motion of particles and bubbles with applications to Coriolis flow meters," *Phys. Fluids*, vol. 20, 2008.

[59] C. F. M. Coimbra, D. L'Espérance, R. A. Lambert, J. D. Trolinger, and R. H. Rangel, "An experimental study on stationary history effects in high-frequency Stokes flows," *Journal of Fluid Mechanics*, vol. 504, pp. 353–363, 2004.

[60] J. R. A. F. Candelier and M. Souhar, "On the effect of the Boussinesq Basset force on the radial migration of a Stokes particle in a vortex," *Physics of Fluids*, vol. 16, pp. 1765–1776, April 2004.

[61] M. Maxey, "The gravitational settling of aerosol particles in homogeneous turbulence and random flow fields.," *Journal of Fluid Mechanics*, vol. 174, pp. 441–465, 1987.

[62] E. Balkovsky, G. Falkovich, and A. Fouxon, "Intermittent distribution of inertial particles in turbulent flows," *Physical Review Letters*, vol. 86, pp. 2790–2793, September 2001.

[63] van Hinsberg, T. Boonkkamp, and H. Clercx, "An efficient, second order method for the approximation of the Basset history force," *Journal of Computational Physics*, vol. 230, February 2011.

[64] A. M. Ardekani and R. H. Rangel, "Unsteady motion of two solid spheres in Stokes flow," *Physics of Fluids*, vol. 18, 2006.

[65] M. Boivin, O. Simonin, and K. D. Squires, "Direct numerical simulation of turbulence modulation by particles in isotropic turbulence," *Journal of Fluid Mechanics*, vol. 375, pp. 235–263, 1998.

[66] S. Sundaram and L. R. Collins, "A numerical study of the modulation of isotropic turbulence by suspended particles," *Journal of Fluid Mechanics*, vol. 379, pp. 105–143, 1998.

[67] M. Pinsky and A. Khain, "Turbulence effects on droplet growth and size distribution in clouds-a review," *Journal of Aerosol Science*, vol. 28, no. 7, pp. 1177–1214, 1997.

[68] S. J. Weidenschilling and J. N. Cuzzi, "Formation of planetesimals in the solar nebula," *Protostars and planets III*, vol. 1, pp. 1031–1060, 1993.

[69] H. Homann, T. Guillot, J. Bec, C. W. Ormel, S. Ida, and P. Tanga, "Effect of turbulence on collisions of dust particles with planetesimals in protoplanetary disks," *Astronomy and Astrophysics*, 2016.

[70] S. Sundaram and L. R. Collins, "Collision statistics in an isotropic particle-laden turbulent suspension. part 1. direct numerical simulations," *Journal of Fluid Mechanics*, vol. 335, pp. 75–109, 1997.

[71] Baw and Peskin, "Some aspects of gas-solid suspension turbulence," *Journal of Basic Engineering*, vol. 94, no. 4, pp. 631–635, 1971.

[72] K. Guseva, U. Feudel, and T. Tel, "Influence of the history force on inertial particle advection: Gravitational effects and horizontal diffusion," *Phys. Rev. E*, vol. 88, 2013.

[73] A. Daitche and T. Tel, "Memory effects in chaotic advection of inertial particles," *Physics of Fluids*, July 2014.

[74] N. A. Fuchs, *The Mechanics of Aerosols*.
Dover Publications, 1989.

[75] C. F. Curtiss and J. O. Hirschfelder, "Integration of stiff equations," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 38, no. 3, pp. 253–243, 1952.

[76] E. Süli and D. F. M. Mayers, *An Introduction to Numerical Analysis*.
Cambridge University Press, 2003.

[77] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical recipes: the art of scientific computing*.
Cambridge University Press, third ed., 2007.

[78] T. Elliott, *Deltas in Sedimentary environments and facies*, pp. 113–154.
Oxford: Blackwell Scientific Publications, 2nd ed., 1986.

[79] O. Sarig, "Introduction to the transfer operator method." Second Brazilian School on Dynamical Systems, October 2012.

[80] P. Cvitanović, R. Artuso, L. Rondoni, and E. Spiegel, *Chaos: Classical and Quantum*.
Niels Bohr Institute, 15.9 ed., June 2017.

[81] S. Klus, P. Koltai, and C. Schütte, "On the numerical approximation of the Perron–Frobenius and Koopman operator," *Journal of Computational Dynamics*, vol. 3, pp. 51 – 79, October 2016.

[82] S. M. Ulam, *A collection of mathematical problems*.
Interscience Publishers, 1960.

[83] G. Froyland, G. A. Gottwald, and A. Hammerlind, "A computational method to extract macroscopic variables and their dynamics in multiscale systems," *Journal of Applied Dynamical Systems*, vol. 13, no. 4, pp. 1816–1846, 2014.

[84] A. Hadjighasem, M. Farazmand, D. Blazevski, G. Froyland, and G. Haller, "A critical comparison of Lagrangian methods for coherent structure detection," *Chaos*, vol. 27, May 2017.

[85] G. Froyland, N. Santitissadeekorn, and A. Monahan, "Transport in time-dependent dynamical systems: Finite-time coherent sets," *Chaos*, vol. 20, November 2010.

[86] T. Sapsis and G. Haller, "Instabilities in the dynamics of neutrally buoyant particles," *Physics of Fluids*, vol. 20, 2008.

[87] B. J. Alder and T. E. Wainwright, "Studies in molecular dynamics. I. general method," *The Journal of Chemical Physics*, vol. 31, pp. 459–466, August 1959.

[88] H. Sigurgeirsson, A. Stuart, and W.-L. Wan, "Algorithms for particle-field simulations with collisions," *Journal of Computational Physics*, vol. 172, pp. 766–807, 2001.

[89] G. Haller, "Distinguished material surfaces and coherent structures in 3D fluid flows.," *Physica D*, pp. 149 – 248, 2001.

[90] S. C. Shadden, *Transport and Mixing in Laminar Flows: From Microfluidics to Oceanic Currents*.
WILEY-VCH, November 2011.

[91] M. Haenggi, "On distances in uniformly random networks," *IEEE Transactions on Information Theory*, vol. 51, October 2005.

[92] S. Elghobashi and G. C. Truesdell, "Direct simulation of particle dispersion in a decaying isotropic turbulence.," *Journal of Fluid Mechanics*, vol. 242, pp. 655–700, 1992.

[93] V. Armenio and V. Fiorotto, "The importance of the forces acting on particles in turbulent flows," *Physics of Fluids*, vol. 13, pp. 2437–2440, 2001.

[94] F. Toschi and E. Bodenschatz, "Lagrangian properties of particles in turbulence," *Annual Review of Fluid Mechanics*, vol. 41, pp. 375–404, 2009.

[95] A. G. W. Lawrie, *On Rayleigh-Taylor mixing: Confinement by stratification and geometry*. PhD thesis, University of Cambridge, June 2009.

[96] V. K. P Ramaprabhu, R. Banerjee, H. Varshochi, M. Khan, and A. G. W. Lawrie, "Evolution of the single-mode Rayleigh-Taylor instability under the influence of time-dependent accelerations," *Physical Review E*, vol. 93, 2016.

[97] R. Courant, K. Friedrichs, and H. Lewy, "On the partial difference equations of mathematical physics," *IBM Journal of Research and Development*, vol. 11, pp. 215–234, March 1967.

[98] P. Bourke, "Interpolation methods," December 1997.

[99] T. Teitelbaum and P. D. Mininni, "The decay of turbulence in rotating flows," *Physics of Fluids*, vol. 23, January 2011.

[100] C. Rorai, D. Rosenberg, A. Pouquet, and P. Mininni, "Helicity dynamics in stratified turbulence in the absence of forcing," *Physical Review E*, vol. 87, June 2013.

[101] D. L. Enrico Calzavirini, Martin Kerscher and F. Toschi, "Dimensionality and morphology of particle and bubble clusters in turbulent flow," *Journal of Fluid Mechanics*, vol. 607, pp. 13–24, 2008.

[102] J. F. Richardson and W. N. Zaki, "Sedimentionation and fluidisation: part 1.," *Transactions of the Institution of Chemical Engineers*, vol. 32, pp. 35–53, 1954.

[103] M. Farazmand and G. Haller, "The Maxey-Riley equation: Existence, uniqueness and regularity of solutions," *J. Nonlinear Analysis-B*, vol. 22, pp. 98–106, August 2014.

[104] J. Bec, H. Homann, and S. S. Ray, "Gravity-driven enhancement of heavy particle clustering in turbulent flow," *Physical Review Letters*, vol. 112, 2014.

[105] J. P. L. C. Salazar, J. D. Jong, L. Cao, S. H. Woodward, H. Meng, and L. R. Collins, "Experimental and numerical investigation of inertial particle clustering in isotropic turbulence," *Journal of Fluid Mechanics*, vol. 600, pp. 245–256, April 2008.

[106] A. J. Petersen, L. Baker, and F. Coletti, "Experimental study of inertial particles clustering and settling in homogeneous turbulence," *Journal of Fluid Mechanics*, vol. 864, pp. 925–970, 2019.

[107] A. Loisy, A. Naso, and P. D. M. Spelt, "Buoyancy-driven bubbly flows: ordered and free rise at small and intermediate volume fraction," *Journal of Fluid Mechanics*, vol. 816, pp. 94–141, March 2017.